

UNIVERSIDAD CARLOS III DE MADRID

ESCUELA POLITÉCNICA SUPERIOR

INGENIERÍA TÉCNICA DE TELECOMUNICACIÓN

TELEMÁTICA



PROYECTO FINAL DE CARRERA

Consumo asociado a las comunicaciones por la interfaz 802.11 del dispositivo Nokia N810 Internet Tablet

Autor: Alejandra García Macías
Tutor: Alberto Cortés Martín

14 de octubre de 2010

Agradecimientos

Siempre es difícil empezar a escribir una página en blanco, numerosas ideas te vienen a la mente, sin saber cuál elegir,... escribir y borrar, hasta que por fin decides empezar.

Sin embargo, en esta ocasión, no se me ocurre mejor comienzo que dando las gracias a mi familia, mis padres, Diego e Isabel, y mi hermano Jose Luis, que sin su apoyo durante estos años, habría sido realmente difícil llegar a donde estoy. Nosotros elegimos nuestro propio futuro, pero tener cerca de ti a personas tan especiales, hace que el camino que en un principio puede parecer de espinas, se convierta en un camino de rosas., lleno de ilusión y de alegría. También, dar las gracias a mi prima Vanessa, que durante este último año me ha aconsejado mucho, y siempre ha estado ahí cuando la he necesitado.

Los agradecimientos parece una de las secciones menos importantes en un proyecto final de carrera, sin embargo, será de una de las más leídas cuando pasen algunos años, y eche la vista atrás. Por eso me gustaría reflejar los primeros años de carrera junto a mis compañeros, destacando en primer lugar, a Joana y a Maria Jose, grandes amigas que todavía perduran, y que siempre que nos juntamos comentamos las vivencias durante estos años de carrera. No olvidar, al resto de compañeros, que siguen presentes y que siempre lo estarán, Alberto, Abel, Natalia, Dani, Silvia, Pablo, Chemi, Mario, Sara, Victor, Adal, Miguel, y que me han hecho más fácil esta etapa.

Dar las gracias también a mis amigas, Rosa, Bea y Eva. Las conozco desde hace muchos años, alguna desde la infancia, y siempre han estado ahí cuando las he necesitado. Una verdadera amistad que durará toda la vida.

Por último, y no significa que sea de menos importancia, dar las gracias a mi tutor Alberto Cortés, por haberme aconsejado durante este año, y por haber tenido mucha paciencia conmigo. Además, dar las gracias a Celeste Durán y a Carlos García, por haberme supervisado también.

Resumen

Nada mas importante para las tecnologías móviles, además de sus prestaciones, que el grado de libertad que puede ofrecernos un dispositivo para mantenernos alejados de las tomas eléctricas y las recargas. La tecnología evoluciona favorablemente pero sin embargo las baterías se resisten a evolucionar de la misma manera que los demás componentes.

Por esta razón, este proyecto trata de minimizar el gasto de la batería optimizando el consumo del interfaz 802.11, basandose en una comunicación wifi en modo infraestructura entre el dispositivo Nokia N810 Internet Tablet y un ordenador.

El escenario planteado permite una comunicación basada en el protocolo UDP para medir el consumo de transmitir y recibir por el interfaz 802.11 en distintos casos según diferentes configuraciones de los parámetros del interfaz.

Abstract

Nothing is more important for mobile technologies, also its benefits, that the degree of freedom that can offer a device to keep away from power points and refills. Technology evolves favorably but the batteries are reluctant to evolve in the same way than the other components.

For this reason, this project tries to minimize the cost, optimizing battery of the 802.11 interface consumption, based on a communication wifi in infrastructure mode between the Nokia N810 Internet Tablet and a computer.

The scenario presented enables a communication based on UDP protocol for measuring power with transmissions and receptions by the 802.11 interface in different cases according to different configurations of the parameters of the interface.

Índice general

1. Introducción	1
1.1. Motivación del proyecto	1
1.2. Objetivos	2
1.3. Contenido de la Memoria	2
 2. Estado del Arte	 5
2.1. Tecnología de comunicacion, Wi-Fi	5
2.2. Nivel de Red, IP	14
2.3. Nivel de Transporte, UDP	15
2.4. Dispositivos Utilizados	16
2.4.1. N810 Internet Tablet	16
2.4.2. N95	19
2.5. Sistemas Operativos	23
2.5.1. Maemo	23
2.5.2. Symbian	24
2.5.3. Comparación: S60 Vs Maemo	25
2.6. Aplicaciones N810	25
2.6.1. Nokia Energy Profiler v1.0	25
2.6.2. Ttcp	29
2.7. Medidas de Energía	30
2.8. Documentos relacionados	34

3. Escenario	39
4. Medidas	45
4.1. Introducción	45
4.2. Estudio del nivel de batería	48
4.3. Power Saving	48
4.4. Unicast	52
4.4.1. Variando tiempo entre paquetes	52
4.4.2. Variando tamaño de paquetes	53
4.5. Multicast	55
4.5.1. Variando el tiempo entre paquetes	56
4.5.2. Variando tamaño de paquetes	58
4.6. Symbian Vs Maemo	59
4.6.1. Unicast	60
4.6.2. Multicast	61
4.7. RTS y CTS	63
4.7.1. RTS=29 y CTS=AUTO	64
4.7.2. RTS=500 y CTS=AUTO	64
4.8. Resultados	65
5. Historia del proyecto	67
5.1. Tareas	67
5.1.1. Tarea 0, Configuración del escenario	67
5.1.2. Tarea 1, Estudio del dispositivo N810 y de la red Wi-Fi . .	68
5.1.3. Tarea 2, Prueba del escenario - Primeras medidas	68
5.1.4. Tarea 3, Medidas	68
5.2. Presupuesto	68
5.2.1. Costes de personal	68
5.2.2. Costes del material	69
5.2.3. Coste total	70

5.3. Problemas encontrados	70
5.3.1. Consumo de la batería cuando el display está apagado . .	70
5.3.2. Compilación de código C en el entorno Maemo	73
5.3.3. Consumo extra del programa Ttcp	74
6. Conclusiones y trabajos futuros	79
6.1. Conclusiones	79
6.2. Trabajos futuros	80
A. Maemo SDK VMware	83
B. Guía de Nokia Energy Profiler v1.0 para Symbian	89
C. Programa Ttcp	95
D. Flasheo e instalación de paquetes en Nokia N810 Internet Tablet	99
E. Cómo realizar las medidas en el Nokia N810 Internet Tablet	103
F. Glosario de Términos	105

Índice de figuras

2.1. Modo ad-hoc	7
2.2. Modo Infraestructura	7
2.3. Proceso de Power Saving	10
2.4. Problema de los nodos ocultos	11
2.5. Linksys WRT54G Wireless Advanced	13
2.6. El entorno OSI	14
2.7. Nokia N810 Internet Tablet	17
2.8. Nokia N95	20
2.9. Comunidad Maemo	23
2.10. Nokia Energy Profiler: Plugin	27
2.11. Nokia Energy Profiler: Statics	28
2.12. Nokia Energy Profiler: Menú	28
2.13. EnergyHub	31
2.14. Efergy eLite	32
2.15. RMS Power Meter	32
2.16. Witricity Power	33
2.17. Google PowerMeter	33
2.18. LocalCooling	34
2.19. Hohm	34
3.1. Escenario	39
3.2. Topología del Escenario	43

4.1. Experimentos	46
4.2. Nivel de la batería	49
4.3. Power Saving	50
4.4. Power Saving en On Intermediate	51
4.5. Unicast, Variando el tiempo entre paquetes	53
4.6. Unicast, Variando el tiempo entre paquetes - Transmisión Vs Re- cepción	54
4.7. Unicast, Variando el tamaño de paquetes	55
4.8. Unicast, Variando tamaño de paquetes - Transmisión Vs Recepción	56
4.9. Multicast, Variando el tiempo entre paquetes	57
4.10. Multicast, Variando el tamaño de paquetes	59
4.11. Symbian Vs Maemo: Unicast, Variando el tiempo entre paquetes .	60
4.12. Symbian Vs Maemo: Unicast, Variando el tamaño de paquetes . .	61
4.13. Symbian Vs Maemo: Multicast, Variando el tiempo entre paquetes	62
4.14. Symbian Vs Maemo: Multicast, Variando el tamaño de paquetes .	63
4.15. Efecto del RTS a 29 y CTS Auto	64
4.16. Symbian Vs Maemo: Efecto del RTS a 29 y CTS Auto	64
4.17. Efecto del RTS y CTS	65
5.1. Error Nokia Energy Profiler	71
5.2. Error Nokia Energy Profiler-conexión a la red cambiando el valor de Power Saving	72
5.3. Ttcp - Consumo extra	76
A.1. Maemo SDK VMWare Appliance v0.6	84
A.2. Scratchbox	85
A.3. Resultado tras ejecutar <i>sb-menu</i> en el scratchbox	86
A.4. Elección del target	86
A.5. Compilación con <i>make</i>	87
A.6. Ejecución con <i>run-standalone.sh ./HelloWorld</i>	87

B.1. Instalación en Symbian Nokia Energy Profiler	90
B.2. Uso en Symbian Nokia Energy Profiler	91
B.3. Formatos para exportar	92
B.4. Archivo exportado (.csv) visualizado con Excel	92
B.5. Iconos de la pantalla de Symbian Nokia Energy Profiler	94
E.1. Maemo, calcular eje del tiempo	104

Índice de cuadros

2.1. Comparación de velocidades de transferencia	5
2.2. Comparación de protocolos de la capa de transporte	16
2.3. Especificaciones Técnicas N810 Internet Tablet ([7])	19
2.4. Especificaciones Técnicas Nokia N95 ([8])	22
2.5. Comparación S60 Vs Maemo	25
5.1. Costes de personal	69
5.2. Costes de material	69
5.3. Coste total	70

Capítulo 1

Introducción

1.1. Motivación del proyecto

A lo largo de los años, la telefonía móvil ha evolucionado a grandes pasos. Comenzó en 1981 donde el fabricante Ericsson lanza el sistema NMT 450 que utilizaba canales de radio analógicos con modulación en frecuencia FM, hasta llegar al presente con la telefonía móvil 3.5G.

Hoy en día, el uso que se le da al teléfono no solo es para hacer llamadas, o comunicarse mediante mensajes de texto o SMS, sino que la utilidad que se le da es completamente multimedia, cámara de fotos, videocámara, internet, videojuegos. Todas estas nuevas prestaciones hacen que la autonomía de la batería del teléfono sea cada vez menor, por lo que reducir el consumo cobra mayor importancia. Por estas razones, es muy importante optimizar las baterías y el consumo de los teléfonos móviles.

Además, un aspecto muy importante que mengua considerablemente la batería, es la conexión wifi. Por ejemplo, un teléfono en reposo, donde la pantalla está apagada, la interfaz inalámbrica consume el 70 % de la potencia total ([28]).

La batería no solo es útil para la telefonía móvil. Se han desarrollado PDAs o Internet Tablets para las cuales la autonomía es muy importante. En el presente proyecto se hace un estudio del consumo asociado a las comunicaciones por la interfaz WLAN en los dispositivos Internet Tablets (Nokia N810).

El objetivo es conseguir que el usuario pueda estar comunicado en todo momento. Esto se consigue gracias entre otras cosas a unas aplicaciones desarrolladas para minimizar el consumo de la batería.

1.2. Objetivos

En el presente proyecto se plantean los siguientes objetivos:

- Estudio del consumo asociado a las comunicaciones por la interfaz WLAN:
 - Cambiando el parámetro de Power Saving del Nokia N810 Internet Tablet.
 - Con transmisiones/recepciones mediante el protocolo UDP.
 - Efecto Unicast/Multicast.
 - Comparación Maemo Vs Symbian.
 - Efecto RTS/CTS.

1.3. Contenido de la Memoria

La memoria está estructura de la siguiente forma:

- El Capítulo 2 introduce las diferentes tecnologías que constituyen el contexto del proyecto. Se revisa la tecnología inalámbrica y se incluye una descripción de los dispositivos Nokia y de sus sistemas operativos, Symbian y Maemo. Finalmente, se describen las herramientas de medición y de comunicación.
- El Capítulo 3 presenta el escenario que se ha utilizado para medir los experimentos del capítulo 4.
- El Capítulo 4 se documentan las pruebas realizadas, incluyendo el resultado y las observaciones de interés.
- El Capítulo 5 describe la distintas fases de desarrollo del proyecto, explicando las tareas realizadas, el presupuesto del proyecto y los problemas encontrados.
- El Capítulo 6 se resumen las principales conclusiones del proyecto y los posibles trabajos futuros.

La memoria dispone tambien de 6 apéndices necesarios para completar algunos aspectos del proyecto:

- El Apéndice A es la documentación sobre el SDK utilizado para la compilación de las aplicaciones llevadas a cabo sobre Maemo.
- El Apéndice B es la guía de la herramienta Nokia Energy Profiler para Symbian.
- El Apéndice C describe la utilización del programa Ttcp utilizado para el envío/recepción de las medidas.
- El Apéndice D documenta cómo flashear e instalar paquetes en el dispositivo Nokia N810 Internet Tablet.
- El Apéndice E muestra cómo se realizan las medidas en en Nokia N810 Internet Tablet.
- El Apéndice F es un glosario de términos utilizado a lo largo del proyecto, mostrando para cada uno su traducción y su significado.

Capítulo 2

Estado del Arte

2.1. Tecnología de comunicacion, Wi-Fi

En esta sección se presenta una visión general de las redes de área local inalámbricas basadas en las especificaciones IEEE 802.11 ([2],[25]) de Junio de 1997.

Su función principal es proporcionar conectividad y acceso como las tradicionales redes cableadas, pero con flexibilidad y movilidad.

El estandar 802.11 ha tenido una evolución constante desde su nacimiento, por esa razón, existen varias versiones que se diferencian entre ellas por la frecuencia en la que actúan además de por la velocidad de transmisión. Este proyecto se centra en la versión 802.11g que opera en la frecuencia 2.4-2.5 MHz con velocidades en condiciones ideales hasta 54 Mbps.

Estándar IEEE	Velocidades de transmisión
802.11 a	54 Mbps
802.11 b	11 Mbps
802.11 g	54 Mbps
802.11 n	600 Mbps

Cuadro 2.1: Comparación de velocidades de transferencia

La interfaz inalámbrica transmite y recibe datos utilizando radiofrecuencia, en lugar del par trenzado, coaxial o fibra óptica utilizado en las LAN convencionales, y proporciona conectividad inalámbrica de igual a igual (peer to peer).

El elemento básico es la estación (STA en el estándar), definida como cualquier elemento que contenga una capa de Control de Acceso al Medio (MAC) y una capa Física (PHY) acorde con lo definido en el estándar. Las estaciones pueden ser móviles, portátiles o estacionarias. En las LANs inalámbricas basadas en el IEEE 802.11 se puede diferenciar otro tipo de elemento habitual, el punto de acceso, que actúa como un concentrador o hub que recibe y envía información vía radio a los dispositivos de usuario.

Un elemento esencial a tener en cuenta en este tipo de redes al utilizarse la radio, es el cifrado. En general se utiliza WEP (Wired Equivalent Privacy), que es un mecanismo de encriptación y autenticación especificado en el estándar IEEE 802.11 para garantizar la seguridad de las comunicaciones entre los usuarios y los puntos de acceso. Otro mecanismo de seguridad definido en el estándar IEEE 802.11 es el conocido como WPA (Wi-Fi Protected Access).

El estándar 802.11 define dos modos operativos:

- Modo ad-hoc. Es una red en la que cada equipo inalámbrico actúa como cliente y como punto de acceso simultáneamente. La configuración que forman las estaciones se llama conjunto de servicio básico independiente o IBSS, la cual tiene al menos dos estaciones y no usa ningún punto de acceso (ver figura 2.1).
- Modo de infraestructura. Cada STA se conecta a un punto de acceso a través de un enlace inalámbrico. La configuración formada por el punto de acceso y las estaciones ubicadas dentro del área de cobertura se llama conjunto de servicio básico o BSS (ver figura 2.2).

Para el desarrollo del proyecto se han tenido en cuenta los siguientes aspectos:

- El dispositivo N810 entra en uno de tres estados, Idle, Transmisión o Recepción.
 - Transmisión:
Dentro del modelo OSI, nos centramos en la capa 2 (Enlace), capa 3 (Red), y la capa 4 (Transporte).
En cuanto a la primera de ellas, cuando una estación quiere transmitir un paquete, debe tener en cuenta si el medio está libre. En este caso, la estación ejecuta una espera llamada espacio entre tramas (IFS).
Si durante ese espacio temporal, el medio sigue libre, cuando termine podrá enviar los datos, de lo contrario, se ejecuta el algoritmo de Backoff, según el cual determina una espera adicional y aleatoria escogida

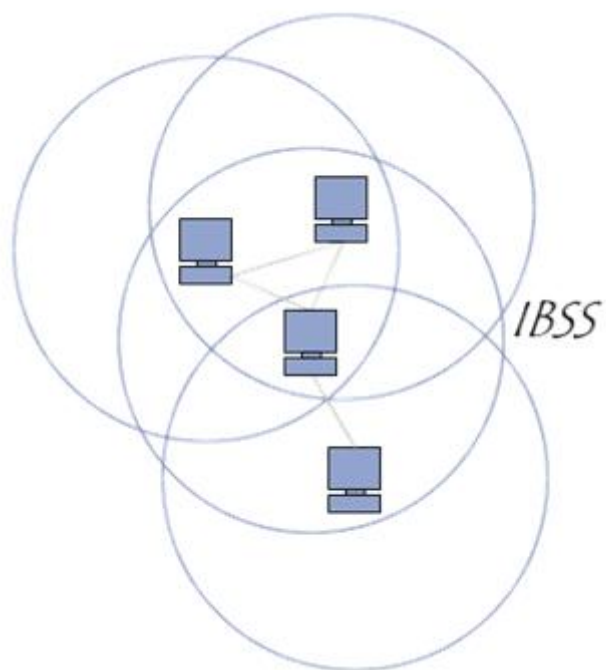


Figura 2.1: Modo ad-hoc

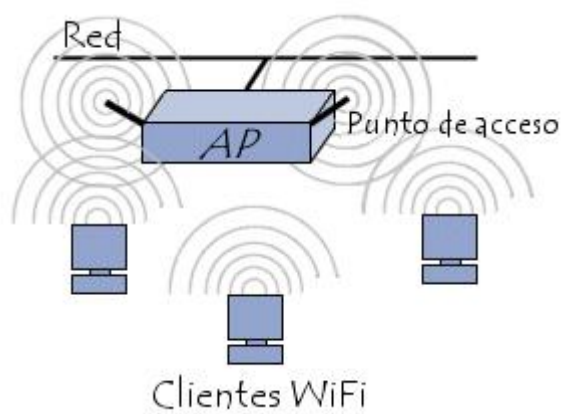


Figura 2.2: Modo Infraestructura

uniformemente en un intervalo llamado ventana de contienda (CW). La función de Backoff es reducir la probabilidad de colisión.

Mientras se ejecuta esta espera, se continúa escuchando el medio de tal manera que si se determina libre durante un tiempo de al menos IFS podrá enviar los datos.

En cuanto a las otras dos capas, tenemos que los datos al enviar van a ir encapsulados en una trama UDP, que a su vez van a ir en una IP. Al tratarse de un protocolo UDP (ver capítulo Nivel de Transporte, UDP 2.3), no se va a establecer la conexión y no se sabe si los paquetes son entregados o no, luego las tramas de control como ACK no van a estar presentes.

Si nos fijamos en el escenario y en los experimentos que se han realizado en este proyecto, tenemos que el medio inalámbrico se encuentra libre, no hay ninguna comunicación más mientras se tomen las medidas, y además, para evitar la fragmentación IP, vamos a evitar sobrepasar el MTU.

- Recepción:

En este caso, la estación lo único que tiene que hacer es recibir los paquetes y escuchar el medio. Durante este período, el medio estará ocupado.

- Idle:

No se envían ni se reciben paquetes, luego el consumo asociado a encapsular los datos y desencapsularlos, al igual que la escucha del medio, no está presente.

- Están presentes distintos tipos de comunicación, punto a punto, punto a multipunto, o broadcast:

- Punto a punto, cada comunicación se establece entre dos nodos.
- Punto a multipunto, se establece entre varios nodos, desde una única ubicación a varios lugares.
- Broadcast, modo de transmisión de información donde un nodo emisor envía información a una multitud de nodos receptores de manera simultánea, sin necesidad de reproducir la misma transmisión nodo por nodo.

- Por otro lado, estas medidas se han tomado según el modelo Unicast y Multicast. Es interesante ver cómo afecta a la batería la comunicación punto a punto o punto a multipunto para luego incorporarlo a la vida real.

Unicast se basa en un proceso de envío de datos desde una máquina origen a una única máquina destinataria o receptor final. Si esa información se quiere enviar a “n” máquinas, habrá “n” comunicaciones punto a punto independientes. De esta forma, en el escenario que se explica en el capítulo 3, se puede verificar que la comunicación unicast tiene lugar entre el N810 y el ordenador.

Por el contrario, Multicast se basa en un único proceso en el que una máquina origen envía a todas las máquinas destinatarias que poseen al menos un miembro de un determinado grupo de multidifusión y que además comparten una misma dirección de multidifusión. A diferencia de Unicast, la máquina origen solo envía una vez la información y no se establecen “n” comunicaciones punto a punto. En este caso son los routers los encargados de realizar las copias de la información y reenviarlas. De esta forma, en el escenario que se puede verificar que la comunicación Multicast tiene lugar gracias a que se ha habilitado en el 802.11g AP y se ha creado un grupo de multidifusión.

Otra diferencia entre ambos modos, son el rango de direcciones utilizado. Así en el primer caso utilizamos una de tipo C (192.0.0.0 a 223.255..255.255), en Multicast se utiliza una de tipo D (224.0.0.0 a 239.255.255.255).

A partir de aquí se comentan distintas características del interfaz 802.11 que se van a modificar en el N810 o en el punto de acceso:

- Las estaciones de red pueden adoptar un modo limitado de potencia. Este modo de funcionamiento implicará que la estación se “despertará” sólo en determinados momentos para conectarse a la red. Estas estaciones (PS-STAs *Power Save Station*) estarán a la escucha de determinadas tramas como la de la portadora y poco más. El control de este tipo de estaciones lo llevará el punto de acceso, que tendrá conocimiento de qué estación se ha asociado en este modo.

El punto de acceso mantendrá almacenado los paquetes que le lleguen con destino a las PS-STAs. Por tanto, el punto de acceso mantendrá un mapa de paquetes almacenados y los destinos a quienes tendrá que repartirlos o enviarlos.

Por lo tanto, en nuestro caso, el parámetro Power Saving consiste en hacer saber al punto de acceso de que el N810 está en modo ahorro de energía. De esta forma, cuando el punto de acceso decida enviarle el paquete lo hará enviándole una trama TIM (*Traffic Indication Map*), a la estación N810 para que despierte. Esto implica un ahorro de energía en el N810 como se puede ver en el capítulo Medidas 4.

La imagen 2.3 muestra un ejemplo de funcionamiento de Power Saving entre el punto de acceso y la PS-STA.

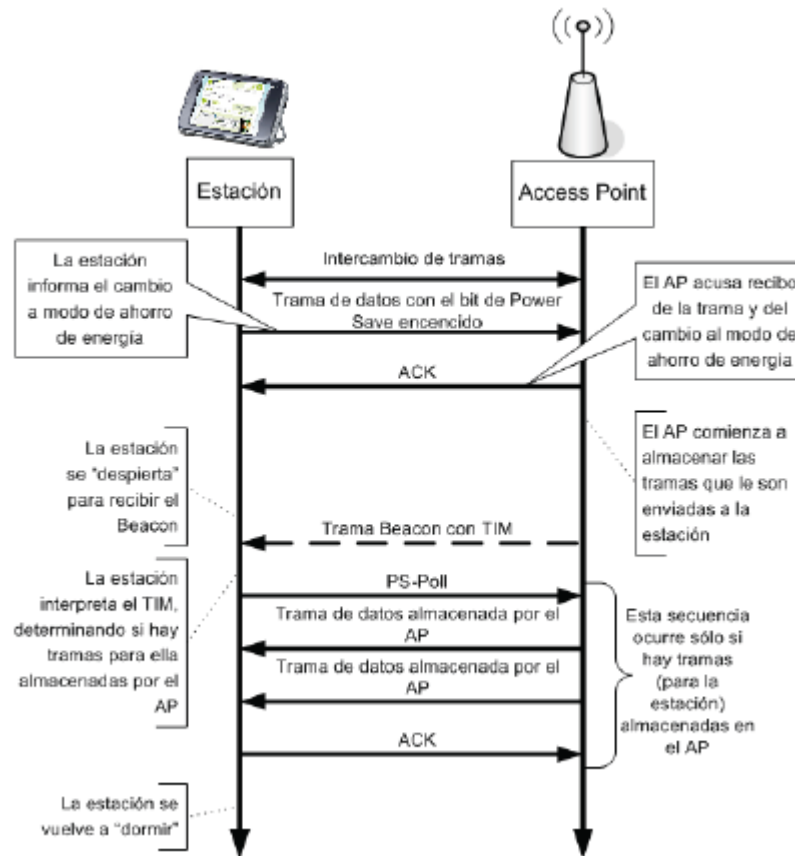


Figura 2.3: Proceso de Power Saving

Este valor se modifica en el N810, tomando 3 posibles valores:

- On Maximum, Power Saving activado.
 - On Intermediate, ahorro de energía intermedio.
 - Off, Power Saving desactivado
- La red inalámbrica tiene fronteras que son difusas donde existen STAs que no se pueden comunicar con otras, algunas veces hasta el punto que no todos los nodos de la red llegan a poder comunicarse con el resto de nodos. Es una situación denominada como problemática del nodo oculto, de la cual se

expone a continuación un ejemplo: tres estaciones A, B y C, las cuales pueden establecer enlaces A-B y B-C, pero las estaciones A y C se encuentran demasiado lejos una de la otra y no pueden recibir sus respectivas señales. En un modo de funcionamiento normal, cuando la estación A transmite, la estación C detectará que el medio está libre y transmitirá (si le interesa), lo que daría lugar a una colisión de ambas tramas en la estación B. La figura 2.4 muestra las estaciones donde A y C están mutuamente ocultos.

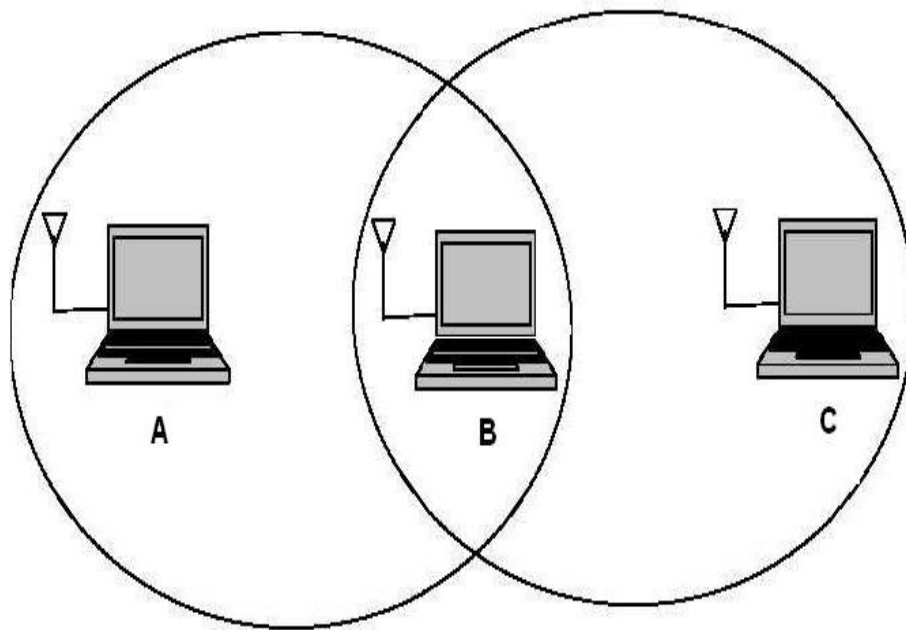


Figura 2.4: Problema de los nodos ocultos

Para solucionar este problema es necesario despejar el área de transmisión utilizando las señales RTS y CTS. Por lo que se ha decidido modificar los valores de RTS Threshold y CTS Protection Mode del punto de acceso.

La trama RTS se usa para reservar el canal para transmitir, al mismo tiempo que silencia a las demás estaciones.

Así, el mecanismo del umbral de RTS proporciona una solución para prevenir colisiones de los datos.

La estación enviará un RTS al punto de acceso, informando que va a transmitir los datos. El punto de acceso responderá con un mensaje de CTS a todas las estaciones dentro de su rango. También confirmará a la estación del solicitante que el punto de acceso la ha reservado para la transmisión

solicitada. El intercambio RTS/CTS tiene lugar para tramas mayores que el threshold. Las tramas menores que el umbral RTS, son enviadas simplemente.

Continuando con el ejemplo anterior de las estaciones A, B y C, al activar el mecanismo RTS/CTS la estación A enviará la trama RTS. Cuando la estación B reciba la trama RTS transmitirá la trama CTS que será recibida por las estaciones A y C. De esta forma la estación C sabrá que se ha reservado el medio y no transmitirá en ese tiempo, proporcionando a la estación A la garantía de que su transmisión estará libre de colisiones.

Los parámetros de la interfaz 802.11 que se pueden modificar en el punto de acceso, son los siguientes:

Authentication Type [*Auto(Default) or Shared Key*] Permite el uso de la autenticación en sistemas abiertos (el remitente y el destinatario no pueden utilizar clave WEP para la autenticación) o Shared Key (el remitente y el destinatario utilizan una clave WEP para la autenticación).

Basic Rate [*Default(Default), All, 1-2 Mbps*] Indica la tasa a la que los clientes se tienen que asociar. La configuración predeterminada asegura la compatibilidad máxima con todos los dispositivos. También se puede optar por permitir todos los tipos de datos mediante la selección de ALL. Para compatibilizar con anteriores dispositivos 802.11 b, hay que seleccionar 1-2Mbps.

Transmission Rate [*Auto(Default), 1 Mbps, 2 Mbps, 5.5 Mbps, 6 Mbps, 9 Mbps, 11 Mbps, 12 Mbps, 18 Mbps, 24 Mbps, 36 Mbps, 48 Mbps, 54 Mbps*] Para especificar la velocidad de transmisión de datos del router¹.

CTS Protection Mode [*Disable(Default), Auto*] CTS ayuda al punto de acceso a mejorar su habilidad de interceptar transmisiones 802.11 g y 802.11 b. Sin embargo, puede disminuir el rendimiento de los dispositivos 802.11 g.

Frame Burst [*Disable(Default), Enable*] Permite enviar paquetes que aumentarán la velocidad de la red, aunque solo se recomienda de 1 a 3 clientes inalámbricos.

Beacon Interval [*Default: 100, Milliseconds, Range: 1 - 65535*] Indica el intervalo de frecuencia del beacon. Un beacon es un paquete broadcast transmitido por el router para sincronizar la red inalámbrica.

¹Dispositivo de interconexión de redes informáticas que permite asegurar el enrutamiento de paquetes entre redes o determinar la ruta que debe tomar el paquete de datos

DTIM Interval [Default: 1, Range: 1 - 255] Indica a los clientes inalámbricos cuál es la siguiente ventana para escuchar mensajes broadcast y multicast.

Fragmentation Threshold [Default: 2346, Range: 256 - 4096] Tamaño máximo para un paquete de datos antes de que se fragmente en múltiples paquetes.

RTS Threshold [Default: 2347, Range: 0-2347] Es el umbral del tamaño del paquete al que se debe enviar el RTS a la unidad de recepción inalámbrica antes de que la unidad de envío inalámbrico abra las comunicaciones.

Tx Antena [Auto(Default), Left, Right] Elegir una antena para transmitir.

Rx Antena [Auto(Default), Left, Right] Elegir una antena para recibir.

Xmit Power [Default: Auto, Range: 0-84 mW] Potencia de transmisión (84 mW no está recomendado)



Figura 2.5: Linksys WRT54G Wireless Advanced

2.2. Nivel de Red, IP

IP (*Internet Protocol*) ([24]), definido formalmente en 1981 en la RFC 791 ([26]), es un protocolo que pertenece a la capa de Enlace (Nivel 3) del modelo OSI ([29]).

Este modelo está basado en una propuesta desarrollada por la ISO (Organización Internacional de Estándares) como un primer paso hacia la estandarización internacional de los protocolos utilizados en varias capas. El modelo se llama OSI (Interconexión de Sistemas Abiertos) de ISO porque tiene que ver con la conexión de sistemas abiertos. Este modelo tiene siete capas (ver figura 2.6), pero nos vamos a centrar en la número 3, Nivel de Red.

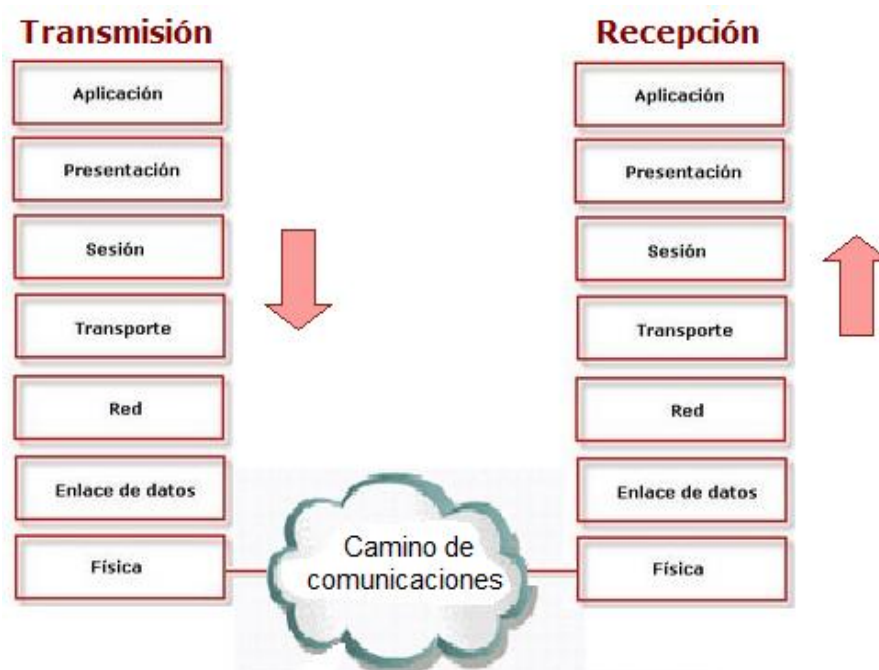


Figura 2.6: El entorno OSI

Esta capa controla las operaciones de la subred (se encarga de llevar los paquetes desde el origen al destino). Para lograr su cometido, la capa de red debe conocer la topología de la subred de comunicaciones y elegir las rutas adecuadas a través de ella.

IP, es un protocolo no orientado a conexión usado tanto por el origen como por el destino para la comunicación de datos a través de una red de paquetes conmutados. Los datos se envía en bloques llamados paquetes o datagramas, que

contienen suficiente información como para que la red pueda encaminarlos hacia el destino, ya que en la cabecera, se incluyen las direcciones de la máquina origen y destino.

2.3. Nivel de Transporte, UDP

En esta sección, nos centramos en la capa 4 (Transporte) del modelo OSI (para saber más acerca de este modelo ver la sección 2.2).

La capa de transporte proporciona un mecanismo para intercambiar datos entre sistemas finales. El servicio de transporte orientado a conexión asegura que los datos se entreguen libres de errores, en orden y sin pérdidas ni duplicaciones. Su función básica es aceptar los datos provenientes de las capas superiores, dividirlos en unidades más pequeñas si es necesario, pasar éstas a la capa de red y asegurarse de que todas las piezas lleguen correctamente al otro extremo.

Hay varios protocolos de la capa de transporte, sin embargo, para la realización de este proyecto, solo se ha utilizado UDP.

UDP (*User Datagram Protocol*) ([19]) es un protocolo descrito en la RFC 768 ([27]). Sus relaciones con los protocolos a nivel de aplicación es escasa, es decir, hay pocos protocolos de nivel 7 que lo emplean como protocolo de transporte. Es cuanto a los de nivel inferior, solo se entiende con el protocolo IP. UDP proporciona un servicio no orientado a conexión para los procedimientos de la capa de aplicación. Así, UDP es básicamente un servicio no fiable, no se garantiza la entrega y la protección contra duplicados. En contrapartida, se reduce la sobrecarga del protocolo.

Esencialmente, incorpora a IP la capacidad de un direccionamiento de puerto, ya que la cabecera UDP incorpora un puerto origen y un puerto destino.

A continuación, se muestra un resumen de las características de los dos principales protocolos de la capa de transporte para explicar las razones por las cuales nos hemos decantado por UDP:

	UDP	TCP
Deteccion de error	sí	sí
Recuperación de error	NO	sí
Control de flujo	NO	sí
Evitar congestion	NO	sí
Orientado a conexión	NO	sí

Cuadro 2.2: Comparación de protocolos de la capa de transporte

Lo que queremos es enviar tramas de nivel de enlace cuando queramos y así controlar el tráfico totalmente, y una manera de hacerlo es con UDP+IP. Es importante que no presente control de flujo ni control de congestión, y así no enviará las tramas cuando sea más eficiente la red.

2.4. Dispositivos Utilizados

Para el desarrollo del proyecto, se ha decidido utilizar dos dispositivos de Nokia con Sistema Operativo diferente, el N95 ([15]) (firmware v11.0.026) y N810 Internet Tablet ([5]) (RX34+RX44 LEGACY R&D version).

La razón de ello, es estudiar el comportamiento y poder hacer una comparativa entre ellos. En mi caso, me he centrado en el N810 Internet Tablet, cuyo Sistema operativo es Maemo (al realizar el proyecto en paralelo con otra compañera, ella se ha encargado del otro dispositivo).

2.4.1. N810 Internet Tablet

Nokia N810 Internet Tablet, pensado para tener Internet en un dispositivo reducido pero sin perder la usabilidad de un ordenador normal, no es un teléfono. El tamaño es de 72x128x14 mm, con una pantalla de 4.13" táctil de 800x480 píxeles. Cuenta con un teclado QWERTY escondido tras la pantalla que ayuda a la hora de escribir tanto e-mails como cualquier dirección web, e incluso tiene funcionalidades como las de un teclado de PC(copiar, pegar, etc) (ver figura 2.7).

El sistema Operativo que utiliza es Maemo OS2008, que al ser software basado en el kernel de Linux, deja la libertad de poder instalar y configurar todo tipo de aplicaciones, como por ejemplo las utilizados para el desarrollo del proyecto, Nokia Energy Profiler y Ttcp (ver capítulo Aplicaciones N810 2.6).

El estándar utilizado para WLAN es IEEE 802.11b/g, de esta forma se puede

conectar a Internet desde cualquier lugar. El único fallo es que al no tener tarjeta SIM no se puede conectar a Internet por 3G, en caso de que no este disponible alguna red Wi-Fi.

Utiliza un navegador basado en Firefox, soportando las tecnologías más utilizadas, incluidas JavaScript,Flash y AJAX.



Figura 2.7: Nokia N810 Internet Tablet

ESPECIFICACIONES TÉCNICAS		
Tamaño	Volumen	128 cc
	Peso	226 g
	Longitud	72 mm
	Anchura	128 mm
	Grosor	14 mm
Pantalla	Pantalla WVGA de 4,1" y alta resolución (800 x 480 píxeles) con hasta 65.000 colores	
Procesador	TI OMAP 2420, 400Mhz	
Memoria	DDR RAM 128MB 256 MB Flash	
Almacenamiento	Hasta 2 GB de memoria interna. Admite tarjetas de memoria miniSD y microSD compatibles (con amplificador). Admite tarjetas de hasta 8 GB. (Las tarjetas SD de más de 2 GB deben ser compatibles con SDHC)	
Tiempos de funcionamiento	Batería	Batería Nokia BP-4L
	Uso continuado (pantalla encendida, LAN inalámbrica activa)	Hasta 4 horas
	Reproducción de música	Hasta 10 horas

	Tiempo siempre online	Hasta 5 días
	Tiempo en espera	Hasta 14 días
Otras características	Carcasa deslizante con teclado QWERTY integrado Receptor GPS incorporado Altavoces estéreo de gran calidad y micrófono de alta sensibilidad Pantalla panorámica de alta resolución Soporte sobremesa integrado Cámara Web VGA integrada Tecla HW para el bloqueo de teclas y pantalla táctil Sensor de luz ambiental	
Conexiones	Estándar WLAN Especificación Bluetooth v. 2.0. +EDR (Perfiles admitidos: HID, FTP) USB de alta velocidad para la conexión con el PC Toma de 3,5 mm para auriculares estéreo (Conector AV Nokia)	IEEE 802.11b/g
Idiomas incluidos	Variantes de teclado HW Idiomas de interfaz de usuario Idiomas de la guía del usuario	inglés, alemán, francés, italiano, español-portugués, escandinavo y ruso inglés británico, francés, alemán, italiano, español, inglés americano, portugués brasileño, francés canadiense, español latinoamericano, danés, sueco, finés, ruso, neerlandés, noruego, portugués inglés británico, francés, alemán, italiano, español, inglés americano, portugués brasileño, francés canadiense, español latinoamericano, danés, sueco, finés, ruso, neerlandés, noruego, portugués, árabe
Sistema Operativo	Fácil instalación de nuevas aplicaciones, subprogramas y personalización Actualizaciones del sistema operativo disponibles en Internet	
Navegación Web	Navegador basado en la tecnología Mozilla con compatibilidad estándar Web avanzada, incluido AJAX Navegación por las páginas mediante desplazamiento, panning o la utilización de las teclas de hardware, ampliación y reducción de los sitios Web Completo plug-in Adobe® Flash® 9 de escritorio, con reproducción de vídeo y sonido	
Media	Reproductor multimedia incorporado para visualizar y escuchar contenidos multimedia descargados, transmitidos ó reproducidos en tiempo real y sencilla administración en el dispositivo de la biblioteca multimedia Acceso directo a medios compartidos a través de Universal Plug and Play (UPnP) Formatos de vídeo admitidos: 3GP, AVI, WMV, MP4, H263, H.264, MPEG-1, MPEG-4, RV (RealVideo) Formatos de audio admitidos: MP3, WMA, AAC, AMR, AWB, M4A, MP2, RA (RealAudio), WAV Formatos de listas de reproducción admitidos: M3U, PLS, ASX, WAX, WVX, WPL	
Comunicaciones	Llamadas con vídeo y mensajes por Internet Aplicación de contactos y presencia automática y sencilla para la centralización de las tareas de comunicación	

	Admite el protocolo de iniciación de sesión (SIP) e interoperatividad con servicios estándar del sector
Mapas	Aplicación de mapas con mapas preinstalados que incluyen lugares de interés Servicios avanzados disponibles para su compra, como el navegador Wayfinder
Correo electrónico	Acceso mediante navegador a los servicios de correo Web más frecuentes Aplicación de correo electrónico para la utilización del correo electrónico personal compatible con IMAP, SMTP y POP3
Imágenes	Visualización de imágenes en pantalla completa y función de secuencia de imágenes Formatos de imagen admitidos: BMP, GIF, JPE, JPEG, PNG, TIF/TIFF, SVG, WBMP
Lector RSS	Lector para la suscripción, administración y actualización con envíos de noticias Web Compatible con RSS 1.0/2.0 y Atom 1.0
Herramientas	Administrador de archivos Lector de PDF Reloj Juegos Chess, Blocks, Mahjong y Marbles Copia de seguridad y recuperación
Métodos de escritura	Teclado HW, teclado táctil en pantalla completa y teclado en pantalla

Cuadro 2.3: Especificaciones Técnicas N810 Internet Tablet ([7])

2.4.2. N95

Nokia N95, es un dispositivo pensado para tener todos los aparatos multimedia, en uno solo. De esta forma, se evita tener que llevar la cámara de fotos, de video, el reproductor de música, la consola para los videojuegos, un dispositivo GPS, y además el teléfono para llamar/SMS independientemente.

Respecto al diseño del teléfono, se trata de un teléfono deslizante, con teclado alfanumérico y panel con pulsadores para la reproducción multimedia. Además incorpora acelerómetro, para que la pantalla se ajuste a la posición que necesite el usuario.



Figura 2.8: Nokia N95

ESPECIFICACIONES TÉCNICAS		
Cobertura	WCDMA 2100 + E850/900/1800/1900MHz Cambio automático de una banda a otra y de un modo a otro	
Dimensiones	Volúmen	96 cc
	Peso	128 g
	Longitud (máx.)	99 mm
	Anchura (máx.)	53 mm
	Grosor (máx.)	21 mm
Funciones de memoria	Memoria dinámica interna de 160 MB* para mensajes, tonos de llamada, imágenes, secuencias de vídeo, notas de la agenda, lista de tareas y aplicaciones. Ranura para tarjeta de memoria que admite tarjetas de memoria microSD de hasta 8 GB	
Alimentación	Batería	Batería Nokia (BL-5F) 950 mAh
	Tiempo en conversación hasta	160 min. (WCDMA), hasta 240 min. (GSM)
	Tiempo en espera	Hasta 200 horas (WCDMA), hasta 225 horas (GSM)
Pantalla	Gran pantalla TFT QVGA (240 x 320 píxeles) de 2,6" con detector de luz ambiental y hasta 16 millones de colores	
Interfaz de usuario	Sistema operativo	Software S60 en sistema operativo Symbian
	Interfaz de usuario	S60 3.1
	Teclas multimedia dedicadas	
	Menú multimedia	
	Pantalla de espera activa	

Administración de llamadas	Contactos	Base de datos de contactos avanzada que admite múltiples datos de teléfono y correo electrónico por registro, también imágenes en miniatura y grupos
	Marcación rápida Registros Rellamada automática Respuesta automática (funciona sólo con kit manos libres portátil o kit de coche compatible) Admite número fijo de marcación, que sólo permite las llamadas a los números predefinidos Llamada multiconferencia Función “Pulsa y habla” (PoC) (consultar la disponibilidad del servicio con el operador)	Guarda la lista de tus llamadas enviadas, recibidas y perdidas
Funciones de voz	Marcación por voz independiente de la persona que habla Comandos de voz Grabadora de voz Tono de llamada de voz Altavoz manos libres integrado	
Mensajería	Mensajería de texto	Admite SMS concatenados, postales electrónicas, listas de distribución de SMS
	Mensajería multimedia Adaptación automática de tus imágenes megapíxel al formato MMS (tamaño máximo: 300 KB, dependiendo de la red) Texto predictivo	Combina imágenes, vídeo, texto y secuencias de audio para enviarlas como MMS a un PC o teléfono Admite los principales idiomas de Europa y Asia-Pacífico
Transmisión de datos	WCDMA 2100 (HSDPA) con voz y paquetes de datos simultáneos (PS velocidad máxima UL/DL= 384/3,6 MB, CS velocidad máxima 64 kbps) Modo de Transmisión Dual (DTM)	Permite la conexión simultánea de voz y paquetes de datos en las redes GSM/EDGE.Clase A simple, multi slot clase 11, velocidad máxima DL/UL: 177,6/118,4 kbits/s EGPRS clase B, multi slot clase 32, velocidad máxima DL/UL= 296 / 177,6 kbits/s
Imaging y vídeo	Cámara de hasta 5 megapíxeles (2.592 x 1.944 píxeles), óptica Carl Zeiss, lente Tessar, grabación de vídeo VGA MPEG-4 hasta 30 fps Conexión directa con TV compatible mediante el Cable de Conectividad de Vídeo Nokia (CA-75U, incluido en el paquete de venta) o LAN inalámbrica/UPnP Cámara delantera, sensor CIF (352 x 288) Capacidad para videollamada y vídeos compartidos (servicios de redes WCDMA) Flash integrado Micrófono estéreo digital Modos de flash	
		Activado, desactivado, automático, reducción de ojos rojos

	Galería giratoria Álbum/Blog online Envío de fotos/vídeos desde la galería Nokia Lifeblog 2.0 support Editores de vídeo y de imágenes fijas Movie Director para la creación automática de vídeos
Vídeo móvil	Resoluciones de vídeo Hasta VGA (640 x 480) a un máximo de 30 fps Grabación de audio AAC mono Estabilización de vídeo digital Duración de las secuencias de vídeo Limitada por la memoria disponible Formato de archivo de vídeo .mp4 (predeterminado), .3gp (para MMS) Balance de blancos Automático, soleado, nublado, incandescente, fluorescente Escena Automática, nocturna Tonos de color Normal, sepia, blanco y negro, negativo, vivo Zoom Digital hasta 10x (VGA hasta 4x)
Especificaciones de la cámara	Sensor CMOS, 5 megapíxeles (2.592 x 1.944) Óptica Carl Zeiss Lente Tessar Distancia focal 5,6 mm Amplitud de enfoque 10 cm infinito Distancia de enfoque macro 10-50 cm Velocidad del obturador Obturador mecánico: 1/1000 1/3 s
Funciones de música	Reproductor de música digital Admite MP3/AAC/AAC+/eAAC+/WMA/M4A con listas de reproducción y ecualizador Altavoz manos libres integrado Compatible con OMA DRM 2.0 y WMDRM para música Radio FM estéreo (87.5-108MHz /76-90MHz)
Visual Radio	Escucha música e interactúa con tus emisoras de radio favoritas Entérate de la canción que está sonando, de su cantante y más información sobre el artista Participa en concursos y responde a encuestas, vota por tus canciones favoritas

Cuadro 2.4: Especificaciones Técnicas Nokia N95 ([8])

2.5. Sistemas Operativos

2.5.1. Maemo

Maemo, es una plataforma de software que se basa principalmente en el código fuente abierto, para dispositivos móviles. Ha sido desarrollada por Nokia en colaboración con muchos proyectos de código abierto como el kernel de Linux, Debian, GNOME, y muchos más ([4]).

Utiliza un sistema operativo Linux basado en una arquitectura ARM ([6]), tratándose de un sistema abierto para desarrolladores voluntarios y empresas que comparten su trabajo gracias a la comunidad Maemo ([3]).

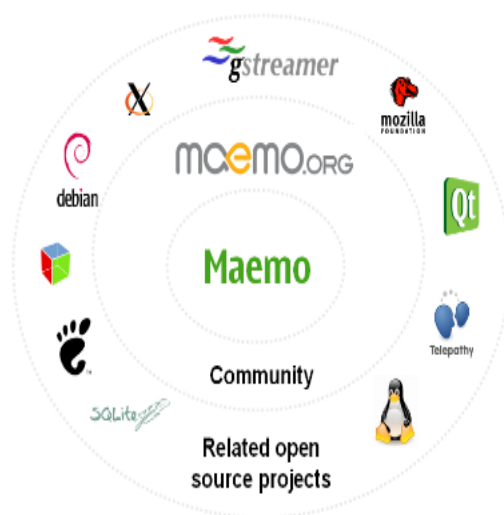


Figura 2.9: Comunidad Maemo

El SDK también se pueden utilizar en otros sistemas operativos a través de un entorno virtual (ver apéndice Maemo SDK VMware A).

Historia:

- 2005: Nokia es la primera empresa en introducir ARM en el desktop de Linux.
El 25 de Mayo anuncia su primera propuesta de Linux, Nokia 770 y la creación de la comunidad Maemo.
- 2006: Código abierto e Internet gracias a la colaboración de ARM, TI, Adobe, Google, Real-Maemo.

Estructura de código como telepathy Gstreamer, y DBUS.
Emergen los dispositivos móviles de Internet (MIDs)

- 2007: Anuncio de los Nokia N800/N810, y de WIMAX.
Emergen smart phones y MIDS con Linux.
Competencia: iPhone, Intel Moblin, Android.
- 2008: Convergencia del chip ARM Cortex A8 y el modem HSPA.
Anuncio de la versión Alpha del SDK de Maemo 5.
- 2009: N900, primer móvil con sistema operativo Maemo 5.

Modelos:

Nokia N770 ([12])

Nokia N800 ([13])

Nokia N810

Nokia N900 ([14])

2.5.2. Symbian

Symbian, es un sistema operativo diseñado para dispositivos móviles y desarrollado por Nokia, ejecutado exclusivamente en procesadores ARM ([16]).

Historia:

- 1998: El 24 de Junio de 1998, se estableció la empresa Symbian Ltd como una alianza entre Ericsson, Nokia, Motorola y Psion para explotar la convergencia entre PDAs y teléfonos móviles.
El teléfono Nokia 7650 ([11]) fue uno de los teléfonos más inteligentes gracias al sistema operativo.
- 2008: El 24 de Junio de 2008, Nokia, Sony Ericsson, Motorola y NTT DO-COMO anuncian su intención de unificar el Sistema Operativo Symbian, S60, UIQ ([18]) y MOAP(S) para crear una plataforma abierta.
- 2009: El 12 de Febrero, 78 socios de la industria apoyan a la fundación Symbian.
Symbian sigue siendo el sistema operativo líder en software móvil con mas de 250 millones de dispositivos a través de más de 250 operadores, y con decenas de aplicaciones de terceros disponibles para estos móviles.

Modelos:

Algunos modelos de compañías como Nokia, Samsung y Ericsson ([10])

2.5.3. Comparación: S60 Vs Maemo

	S60	MAEMO
SERVICIO	Variedad de servicios móviles y de internet	Variedad de servicios de Internet, como OVI ([17])
INTERFAZ DE USUARIO	Interfaz de Usuario para la gran variedad de dispositivos móviles con los factores de que la forma, métodos de entrada y el tamaño de la pantalla son distintos	Interfaz de usuario optimizada para pantallas táctiles de gran tamaño
NAVEGADOR WEB	Webkit con Flash Lite 3	Engine y JavaScript optimizados de Firefox 3.1 + flas 9.4
PERSONALIZACIÓN	Personalización con elementos UX, servicios y aplicaciones	Gran variedad de widgets
MIDDLEWARE Y LLIBRERIAS	Frameworks específicos, y código abierto como Webkit	Frameworks compartidos con Moblin (Linux MID) y Ubuntu SW (Linux desktop), 80 % código abierto
SISTEMA OPERATIVO	Symbian. Optimizado para dispositivos móviles, robusto y seguro	Linux
HARDWARE	Gama Media	Alto rendimiento. Altos procesadores (>500 MHz)

Cuadro 2.5: Comparación S60 Vs Maemo

2.6. Aplicaciones N810

Para el desarrollo del proyecto, se ha utilizado dos herramientas fundamentales:

- Nokia Energy Profiler, que se encarga de medir la potencia consumida por la batería.
- Ttcp, que se encarga de la transmisión/recepción de los paquetes según unos patrones configurables.

2.6.1. Nokia Energy Profiler v1.0

Nokia Energy Profiler, permite gestionar y llevar el control del consumo de energía hecho por la batería del dispositivo. Es una herramienta diseñada para

los desarrolladores que les permite probar y supervisar su aplicación en tiempo real y así realizar aplicaciones más eficientes. El programa opera en dos modos, Grabación y Visualización. En el primer modo, se pueden tomar medidas, salvarlas e incluso exportarlas a formatos como .csv, png o mediante screenshots. El modo de visualización permite mostrar unas medidas guardadas previamente. Está disponible en los Sistemas Operativos Symbian y Maemo.

Symbian

Para este Sistema Operativo, se puede descargar gratuitamente de la página de Forum Nokia ([9]), y consiste en un fichero SIS para ser instalado en el dispositivo móvil. La herramienta se puede instalar en todos los S60 3^a Edición, aunque el modo de grabación solo funciona en FP1 (Feature Pack 1) hacia adelante. Aunque actualmente se encuentra en la version 1.1, para el desarrollo del proyecto se ha utilizado la versión 1.0. En Forum Nokia ([21]) se puede encontrar una guía actualizada sobre esta herramienta o bien, desde su propio menú. La guía de la versión 1.0 se muestra en el apéndice B.

Maemo

En este caso, se ha utilizado una herramienta facilitada por Nokia Finlandia, pues no está disponible para descargarlo desde la web. Además, al contrario que para Symbian, no existe ninguna guía pública sobre la herramienta, solo se puede consultar su propio menú.

El funcionamiento es parecido al de Symbian, solo que incluye los siguientes cambios:

1. Al comenzar a tomar medidas, el interfaz es distinto que el Energy Profiler 1.1 de Symbian, puesto que al arrancar la herramienta aparece la ventana en blanco, y hay que ir a “Menu >> Plugins >> Power”. De esta forma, se muestra la misma ventana que con Symbian. Ahora habría que pulsar “Menu >> Start”.
2. La ventana que muestra el consumo en tiempo real, tiene menos símbolos que la de Symbian. Consta de:
 - a) Línea vertical blanca: Comienzo y final de la región de medidas capturadas.
 - b) Línea vertical roja: Indica los screenshots tomados y que se pueden visualizar

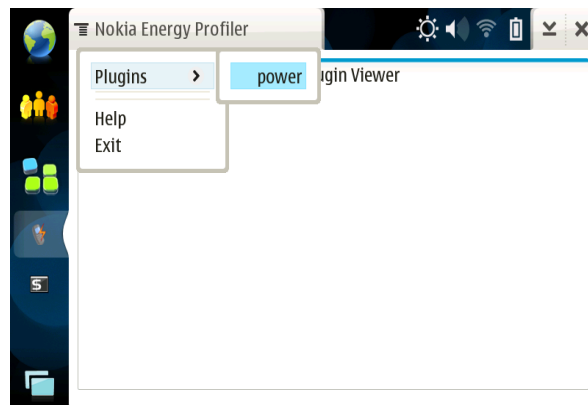


Figura 2.10: Nokia Energy Profiler: Plugin

c) Línea horizontal blanca: Media del consumo en la región.

d) Ejes

1) X: Tiempo [s]

2) Y: Consumo [W]

3. Menú:

a) Plugins: Selecciona el Plugin para empezar a tomar medidas

b) Start: Empieza a tomar medidas

c) Stop: Para de tomar medidas

d) Clear: Limpia la ventana de datos.

e) Settings: Para establecer el periodo de medición y para activar el screenshot

1) Battery capacity (mAh) [Fixed - This Phone].

2) Power(seconds)

- Measurement period (0.25, 1, 5)

- Screenshot trigger (0-10000)

f) File

1) Load: Para cargar las medidas (.jce)

2) Save: Para guardar las medidas (.jce)

3) Export: Exportar las medidas a csv, imágenes png, y screenshot.

g) Statics: Información de estadísticas

- 1) Box Plot
- 2) Histogram
- 3) Summary

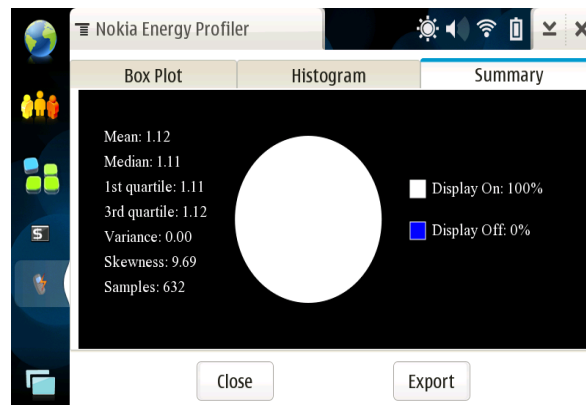


Figura 2.11: Nokia Energy Profiler: Statics

- h)* Close: Cierra la ventana de medidas
- i)* Help: Muestra la ayuda
- j)* Exit: Cierra la herramienta

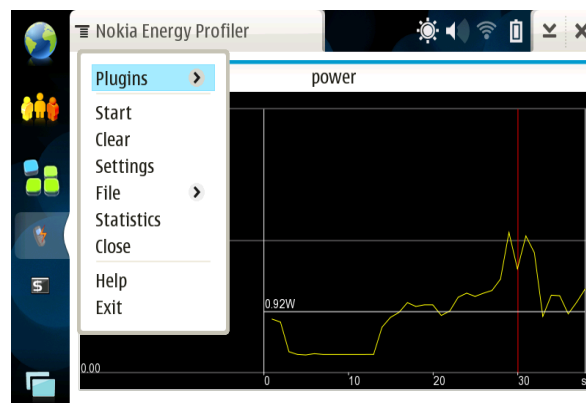


Figura 2.12: Nokia Energy Profiler: Menú

4. Teclado: En el caso de Maemo, dispone de menos teclas:

- a)* Selección: Alterna entre el desplazamiento horizontal y modo de zoom

- b) Arriba: Aleja el eje Y
 - c) Abajo: Acerca el eje Y
 - d) Izquierda: Aleja el eje X o se desplaza hacia la izquierda
 - e) Derecha: Acerca el eje X o se desplaza hacia la derecha
5. A la hora de exportar, algunos datos no estan todavía disponibles en el csv, como es el caso de la corriente y Voltaje.

2.6.2. Ttcp

TTCP, es un programa basado en sockets² que mide el rendimiento de procesamiento de la red entre dos sistemas. Originalmente fue desarrollado para el sistema operativo BSD en 1984.

Contabiliza el tiempo de transmisión y recepción de datos entre dos sistemas que usan los protocolos TCP o UDP. La versión utilizada es la 1.12 (año 2002) que incluye una nueva opcion para fijar el tiempo entre paquetes.

Ttcp funciona en dos modos - Recepción y Transmisión. En el modo de recepción, el proceso es lanzado por ejemplo en un ordenador cuando se quiere transmitir desde el dispositivo N810. En el modo de transmisión, el proceso de envío se pone por ejemplo en el dispositivo, mientras que en el ordenador se lanza el proceso de recepción. Siempre hay que lanzar primero el proceso de recepción antes que el de transmisión.

Las medidas han sido tomadas para modos unicast y multicast, además de utilizar entre otras opciones -u (UDP) y -w (tiempo entre paquetes).

Usage:

```
ttcp -t [-options] host [ < in ]
ttcp -r [-options] [multicast-group][ > out]
```

Ttcp genera un consumo propio debido a un proceso interno por la llamada al sistema, select(). Esta función permite fijar un tiempo máximo de espera, y se incluye en el programa con la nueva version (1.12) debido a la opcion -w. Este consumo es aproximadamente de 0.2W (ver apartado Problemas Encontrados, Consumo extra del programa Ttcp, 5.3.3).

²Mecanismos de comunicación entre procesos que permiten que emita o reciba información con otro proceso incluso estando en distintas máquinas. Esta interfaz de comunicaciones es una de las distribuciones de Berkeley al sistema UNIX, implementándose las utilidades de interconectividad de este Sistema Operativo (rlogin, telnet, ftp, ...)

Para ver más sobre esta herramienta consultar el apéndice C.

2.7. Medidas de Energía

La Energía tiene diversas definiciones, relacionadas con la idea de una capacidad para obrar, transformar o poner en movimiento. La que aquí nos interesa es la que se define en la física: capacidad para realizar un trabajo, el cual se mide según el Sistema Internacional de Unidades en julios.

Según esta definición, se obtiene la siguiente fórmula: $W = P * t$, siendo W el trabajo (que equivale a energía, ya que $1J = 1W*s$), P la potencia, y t el tiempo. Por lo tanto, la potencia dice cómo de rápido consume la energía un aparato. De esta forma, también es común dejarlo en “(unidad de potencia) multiplicado por (unidad de tiempo)”: si se mide el trabajo en kW (kilowatio, 1000 W), y el tiempo en horas, la medida resultante es el kWh.

En general, el funcionamiento de los dispositivos para medir la energía constan de dos secciones: Analógica y Digital. Tomando como ejemplo el Multímetro, el circuito interno de la primera sección consta de divisores de tensión y corriente de entrada, el conversor de corriente alterna a continua, el conversor de resistencia a voltaje continuo, la tensión de referencia para comparación, la fuente de alimentación, el detector de pico y las llaves de selección de rango³ y funciones⁴. La sección Digital esta compuesta por el conversor Analógico a Digital y una pantalla de dígitos, que puede ser de Led o de Cristal Liquido.

Sin embargo, a la hora de medir la energía consumida, existen distintos factores como la precisión, exactitud,... que hacen que los instrumentos convencionales basados en el valor medio sean del todo ineficientes. Aún así, a continuación se muestra una lista de hardware o software para medir la energía:

- **Hardware:** Hay diversos aparatos que proporcionan el consumo de energía, ya sea para el hogar, como para medir un circuito, o la potencia en una red eléctrica. A continuación, se muestra un ejemplo dependiendo de su función:

Tester o multímetro, instrumento de medición que ofrece la posibilidad de medir distintos parámetros eléctricos y magnitudes en el mismo dispositivo. Las funciones más comunes son las de voltímetro, amperímetro y

³Elementos que toman una muestra reducida de la señal a medir, a fin de poder procesar esa muestra dentro de los niveles de tensión que manejan los circuitos electrónicos. Estos llaves de rangos también dan el rango de la escala en uso en ese momento.

⁴Dirigen las señales sensadas hacia los distintos circuitos conversores según corresponda para cada una de las aplicaciones

óhmetro. Con esta herramienta, podemos medir el consumo de energía de una gran variedad de dispositivos.

EnergyHub, dispositivo que ayuda a medir y controlar el consumo de energía en el hogar. Gracias a su pantalla táctil LCD desde la cual se puede acceder a todos los datos de consumo de energía por artefacto y por lugar de la casa. Desde luego, también se puede conocer el consumo instantáneo y acumulado de nuestro hogar. Además, estos aparatos evolucionan, e incluso permite acceder a las mediciones de consumo a través de internet, y también compararlas con las de nuestros vecinos, u otras personas que también lo utilicen, para saber si estamos dentro de lo recomendable (ver figura 2.13).



Figura 2.13: EnergyHub

Efergy eLite, dispositivo para el hogar que permite medir tanto el consumo como saber en todo momento cuanto habrá que pagar a partir del precio por kw consumido. Además dispones de función de historia para poder seguir mejor tus consumos, como visualizar los días anteriores uno a uno (ver figura 2.14).

RMS Power Meter, para medir un aparato eléctrico, como un ordenador, hay que conectar este dispositivo entre el aparato y el enchufe, y da la tensión, corriente, potencia y factor de potencia en cada momento (ver figura 2.15).

- **Software:** Al igual que en el anterior caso, existe gran variedad de software para controlar la energía que se consume. Como ya se ha visto en el apartado 2.6, Nokia Energy Profiler, es un software para dispositivos Nokia que permite saber la potencia en cada instante, además de poder sacar



Figura 2.14: Efergy eLite



Figura 2.15: RMS Power Meter

gráficos, medias,...

Witricity Power, se usa para calcular el consumo de energía de diferentes dispositivos eléctricos como ordenadores portátiles, ordenadores, teléfonos móviles y muchos otros productos eléctricos que se usan en el hogar. El programa permite seleccionar el dispositivo para el que se desea controlar el consumo de energía y muestra diferentes informes , tales como cargo por hora, cargo mensual, consumo mensual (ver figura 2.16).

Google PowerMeter, servicio dirigido al consumo de energía en el hogar, con el que se recopila la información de diferentes medidores de consumo, de forma que se pueden analizar los datos y ver cómo se puede ser más eficiente energéticamente (ver figura 2.17).



Figura 2.16: Witricity Power

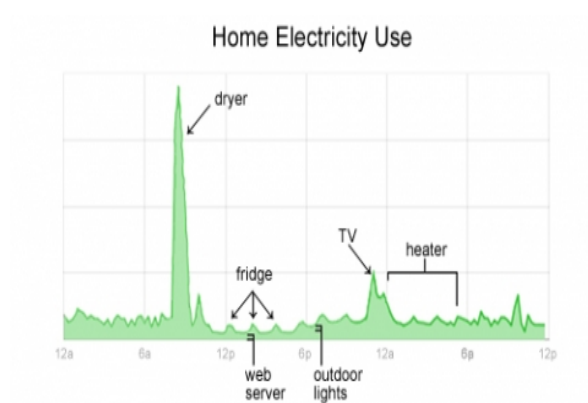


Figura 2.17: Google PowerMeter

LocalCooling, aplicación gratuita que corre sobre XP o Vista para optimizar el consumo de energía de los ordenadores de una forma efectiva. Aunque no modifica nada que no pueda configurarse en el panel de control (suspensión de monitor, discos y procesador) ofrece un control unificado y práctico, además de información interesante. Activado en la barra de tareas, despliega una pequeña ventana que informa el ahorro de Kwh y su equivalente en ahorro de petróleo y árboles (ver figura 2.18).

El sitio de LocalCooling nos recibe con un comunicado alentador: *“Hasta ahora, nuestros 256215 usuarios han ahorrado 2705110.0 Kwh., que equivalen a 1298452.8 galones de petróleo o 143370.83 árboles”*.

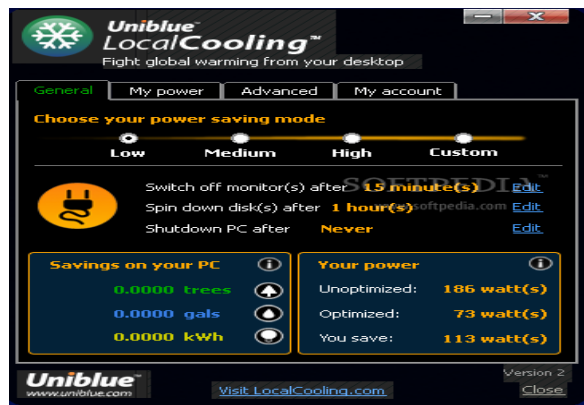


Figura 2.18: LocalCooling

Hohm (Microsoft), aplicación de utilización online con la intención de entregar un servicio a los usuarios para monitorear con él sus gastos energéticos, tanto en el hogar como en la oficina (ver figura 2.19).



Figura 2.19: Hohm

2.8. Documentos relacionados

Debido a que los dispositivos móviles dependen de la batería, es importante minimizar su consumo, lo cual, además es muy importante para los desarrolladores de protocolos. Hay numerosos documentos e investigaciones donde se expone el

tema del consumo y los cuales se han leído previamente a la realización de este proyecto:

- Un ejemplo de ello es el documento de L. Feeney and M. Nilson ([23]), donde se describe una serie de experimentos que obtuvieron mediciones detalladas del consumo de energía de la interfaz de red IEEE 802.11 en un entorno de redes ad hoc.

Se ha llegado a la conclusión de que el consumo en la interfaz 802.11 se comporta de distintas maneras, siendo muy relevante para el diseño de los protocolos de la capa de red; el consumo de energía no es sinónimo al ancho de banda utilizado.

Las medidas tomadas del consumo de energía en transmisión, recepción y descarte de paquetes de varios tamaños, se presentan según un modelo lineal, siendo lo que se esperaba del protocolo IEEE 802.11 ad hoc, aunque la señal sea ruidosa y se halla hecho lo posible para evitar el tráfico en el canal de prueba:

$$\text{Energía} = m \times \text{size} + b$$

Dependiendo de las medidas, el coste de incremento (m) y el fijo (b), varían.

El consumo de energía asociado a recepción no es insignificante y el coste fijo es muy elevado. La transmisión consume menos por paquete, ya que el protocolo MAC y el tráfico broadcast utilizan menor tasa de transmisión.

Cuando se opera en modo ad hoc, el consumo de energía en modo Idle es significativo, los hosts deben mantener sus interfaces de red en reposo con el fin de cooperar para mantener el routing de la red.

Sin embargo se presentan varios inconvenientes: los cambios de asociación de un cluster a otro pueden exigir un procesamiento adicional que aumente el consumo del equipo y los equipos deben mantener la interfaz inalámbrica inactiva todo el tiempo.

- De la misma autora, L. Feeney, nos encontramos con otro artículo ([22]). Este trabajo confirma la hipótesis de que la utilización de recursos en los protocolos de enrutamiento en redes móviles ad hoc (manet) no es contemplada por evaluaciones que tengan en cuenta sólo el ancho de banda.

Manet es un grupo de móviles que cooperan para formar una red independiente de cualquier infraestructura fija o centralizada. En particular, manet no utiliza estaciones, un nodo se comunica directamente con los nodos que

pertenecen a la misma cobertura inalámbrica, e indirectamente con todos los demás nodos utilizando la ruta multi-hop.

El trabajo presentado, define un modelo de consumo de energía que es conveniente para el uso con el tráfico de datos y la movilidad en redes ad hoc. En conclusión:

- El entorno ad hoc impone costes significativos para la capa de enlace.
 - El consumo de energía del tráfico de la recepción y de los descartes, es a la vez sustancial y desproporcionado respecto al número de paquetes o bytes transmitidos.
 - Debido a que el coste del tráfico de los descartes es bajo en relación con el consumo de energía en modo Idle, el coste de funcionamiento de la interfaz en modo promiscuo es alta.
 - El algoritmo de inundación para conocer la ruta de los nodos, tiene un alto coste.
 - El coste incremental asociado con un paquete es relativamente pequeño en comparación con el coste fijo.
 - De los protocolos de enrutamiento DSR o AODV, el más eficiente para transmitir y recibir es el primero.
- Por otro lado, tenemos el documento de Ashima Gupta y Prasant Mohapatra ([20]). Este documento ofrece un estudio detallado del consumo de energía de varios componentes de los teléfonos wifi.

Con el despliegue y uso de estos dispositivos han surgido varias cuestiones que deben abordarse a fin de sostener su crecimiento en el mercado. Los temas principales incluyen la capacidad, seguridad, calidad y potencia.

El propósito de este documento es estudiar las características de consumo de energía de diversos aspectos y componentes en los teléfonos wifi. Por lo tanto, es importante reducir al mínimo el consumo de energía, manteniendo la calidad de servicio requerido. Las medidas tomadas en el estudio son realizadas con un terminal VoWifi (voz sobre wifi).

Las medidas son las siguientes:

- Variando el período Beacon: A medida que disminuye el período, aumenta el consumo (mA). Debido a que el dispositivo se pierde muchas tramas beacons y por lo tanto se queda despierto para recibir las siguientes.
- Scanning de la red: Si se aumenta el periodo de escaneo, disminuye el consumo (mA). A partir de los 200 ms se estabiliza.ç

- DTIM: Si se aumenta, disminuye el consumo, pero produce una disminución de los beneficios. Se llega al máximo ahorro con $DTIM = 10$ (En el estudio se utiliza 2).
- Según el estado del dispositivo: Cuando se encuentra en “Llamada”, consume más que en “Standby”. Por otro lado, a valores más bajos de RSSI, se consume menos que siendo altos.
- Se añade otro punto de acceso: Cuando la señal es baja o el dispositivo esta en “Llamada”, y se ha añadido otro punto de acceso, el consumo del dispositivo disminuye. Sin embargo, cuando la señal es alta o se encuengra en modo “Standby”, el consumo no varía.
- La diferencia entre Ps-Poll y UAPSD es despreciable.

Se determina que los algortimos de exploración tienen un impacto significativo sobre la duración de la batería de los dispositivos VoWifi.

Aunque UAPSD no proporciona mucho ahorro de energía sobre PsPoll, puede disminuir el tráfico del punto de acceso y causar menos contención, aumentando así su capacidad

Capítulo 3

Escenario

En este capítulo se presenta el escenario necesario para realizar las medidas de los experimentos del capítulo 4.

El escenario utilizado está en modo infraestructura porque además del terminal inalámbrico, está presente un punto de acceso (ver imagen 3.1).

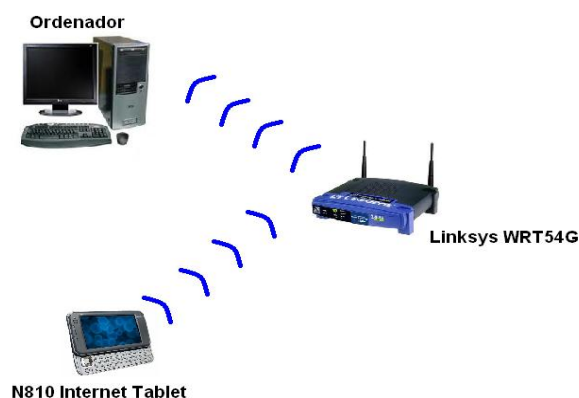


Figura 3.1: Escenario

La principal razón por la que se ha elegido este tipo de escenario es porque se quiere ver el efecto de la configuración del punto de acceso en el consumo de energía.

Sin embargo, el escenario anterior se ha modificado ya que a la hora de escuchar la red inalámbrica para ver los paquetes enviados/recibidos, éstos van aparecer duplicados porque pasan por el punto de acceso. Por este motivo, se ha decidido conectar el ordenador y el punto de acceso por cable ethernet.

Por otro lado, los ordenadores se han conectado a internet solamente para poder realizar las instalaciones de los paquetes básicos del sistema operativo Linux. Internet no es necesario para realizar los experimentos.

A continuación se presenta el hardware y la configuración de la conexión para el escenario. Cabe destacar que al tratarse de un proyecto que se ha realizado con otro en paralelo cuyos experimentos son los mismos que los presentados en el capítulo 4 sólo que con distinto terminal inalámbrico y sistema operativo, el escenario se trata del mismo. Por esta razón, se presenta el escenario conjunto para ambos proyectos (ver figura 3.2):

1. HARDWARE

a) Dos ordenadores

■ Lindo

- 802.3 NIC0: eth0
- 802.3 MAC0: 00:04:75:ca:8b:ae
- 802.3 NIC1: eth1
- 802.3 MAC1: 00:c0:df:08:5c:77
- 802.11 NIC: wifi0
- 802.11 MAC: 00:04:75:bb:65:77
- SSH RSA Fingerprint: c4:a2:38:3d:cc:d2:1c:79:cf:ec:95:72:95:48:92:47
- Kernel: Linux 2.6.18-6-686
- Usuarios(miembros de la practica, tutores y becarios): alcor-tes, alejandra, celeste, demo, estrella

■ Dahl

- 802.3 NIC0: eth0
- 802.3 MAC0: 00:40:f4:be:48:b1
- 802.3 NIC1: eth1
- 802.3 MAC1: 00:04:75:ca:90:42
- 802.11 NIC: wifi0
- 802.11 MAC: 00:04:75:bb:65:75
- SSH RSA Fingerprint: 0e:95:51:ce:9f:9d:bf:a0:7a:92:b5:f4:a3:46:b1:0c

(el resto es igual que el anterior)

b) N95 (Symbian)

c) N810 (Maemo)

d) Ethernet HUB

e) 802.11g Punto de Acceso

- Modelo: WRT54G
- Modo: gateway
- 802.3 MAC: 00:16:B6:2B:6D:14
- 802.11 BSSID: 00:16:B6:2B:6D:16
- 802.11 SID: perlab-nokia-00
- 802.11 Security: no
- 802.11 Authentication: no

Los parámetros del punto de acceso se pueden cambiar autenticándose en el interfaz http. Si se desconocen los valores así como la dirección IP, basta con resetearlo (apretando el botón reset de la parte de atrás del punto de acceso unos 4 segundos) y se obtendrán los parámetros por defecto:

User: admin
Password: admin
IP: 192.168.1.1

2. CONFIGURACIÓN DE LA CONEXIÓN

Ambos equipos están conectados al Ethernet HUB a través de uno de sus 802.3 NIC (eth0). El uplink del HUB está conectado con la red del departamento, y a su vez con internet.

Los equipos también están conectados a una red privada a través del punto de acceso y sus Ethernet NICs. Para conectarse a la red 802.11 se utilizará el ESSID proporcionado por el punto de acceso.

El archivo de los interfaces `/etc/network/interfaces` de los ordenadores es el siguiente:

```
auto lo eth0 eth1
iface lo inet loopback

# The primary network interface
iface eth0 inet dhcp

iface eth1 inet static
    address 192.168.1.81
    netmask 255.255.255.0
```

```
iface wifi-local inet static
    address 192.168.1.181
    netmask 255.255.255.0
    wireless-essid perlab-nokia-00
```

```
iface wifi-uc3m inet dhcp
    wireless-essid Wi-Fi-UC3M
```

```
iface wifi-wlit inet dhcp
    wireless-essid WLIT
    wireless-key *****
```

Para conectarse a Internet, se utiliza el servicio DHCP que se usa en nuestro caso en la red informática del departamento IT.

Se abre el interfaz eth0:

```
sudo ifup eth0
```

Para conectarse a la red local, wifi, diseñada en el escenario, se abre el interfaz eth1:

```
sudo ifup eth1=local
```

Para conectarse a la red local, wifi, utilizando el interfaz inalámbrico:

```
sudo ifup wifi0=wifi-local
```

Nota: Para utilizar una red de backup, en nuestro caso se trata de una red 802,11 cuyo ESSID es WLIT y utiliza clave WEP, hay que hacer lo siguiente:

```
sudo ifup wifi0=wifi-wlit
```

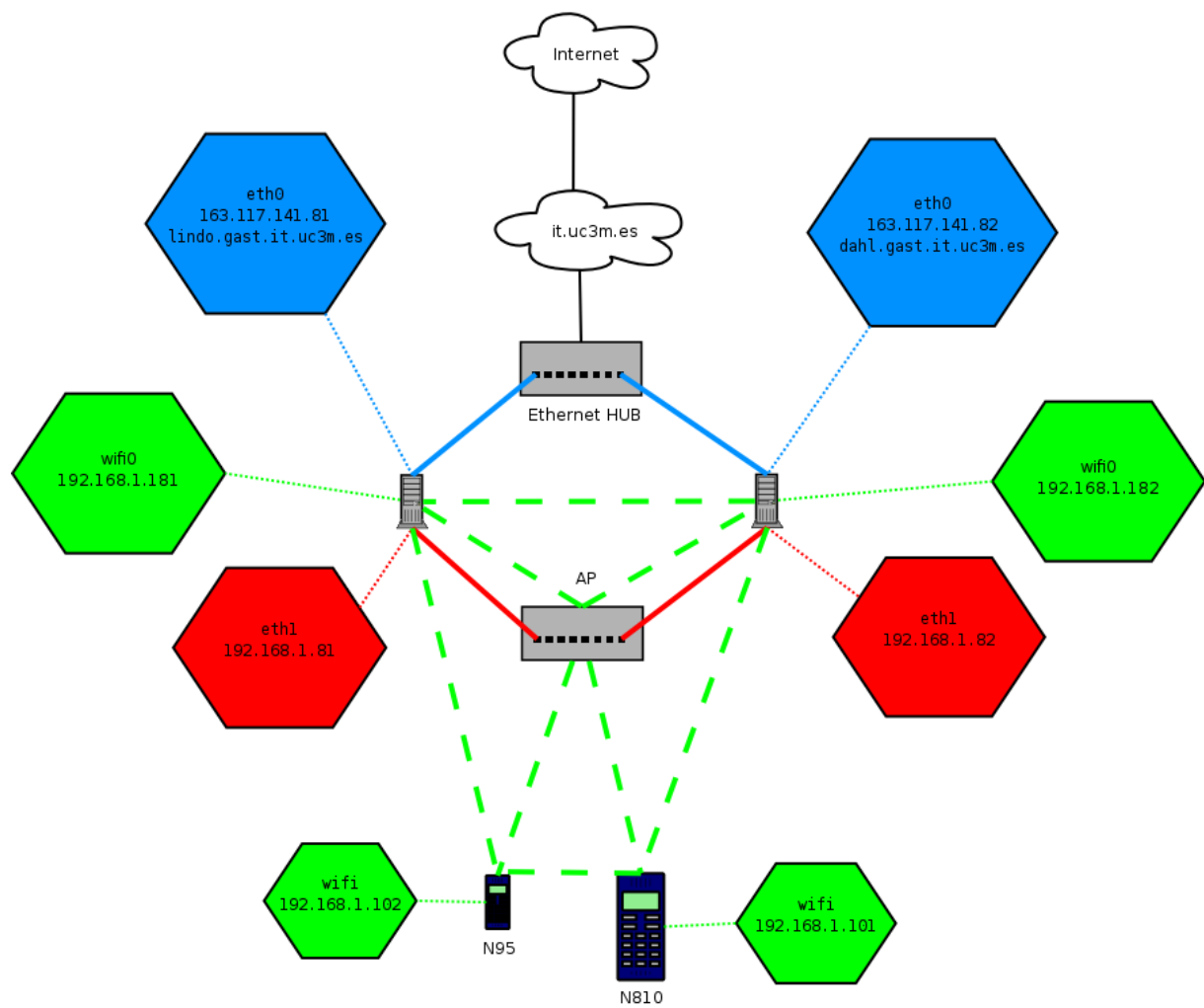


Figura 3.2: Topología del Escenario

Capítulo 4

Medidas

4.1. Introducción

Este capítulo se centra en los experimentos que se han realizado para el estudio del consumo asociado a la interfaz 802.11 del Nokia N810 Internet Tablet. En el capítulo 2.1 se explica en detalle las diferentes especificaciones de la interfaz.

En paralelo a este proyecto se ha realizado otro tomando las mismas medidas pero con distinto terminal y sistema operativo, Nokia N95 (Proyecto Final de Master de Estrella García), por lo que además de las medidas para el N810 habrá una subsección con la comparación entre ambos proyectos.

Los experimentos han sido tomados variando dos parámetros, “periodo entre paquetes” y “tamaño de paquete”. La figura 4.1 muestra una relación entre ambos, y los límites entre los que se van a tomar las medidas.

Por un lado, tenemos como límite superior en “tamaño de paquete”, 1472, que es el tamaño máximo de payload para evitar fragmentación. Por otro lado, tenemos como límite inferior en “periodo entre paquetes”, 32, que es a la máxima velocidad a la que el N810 puede mandar paquetes de 1472 bytes. Para definir los dos límites restantes, hemos elegido una tasa aleatoria de 6,8 Kbps que nos parecía adecuada.

Por lo tanto, las medidas se van a tomar en el plano que queda comprendido entre esos límites, y que a demás siguen las siguientes funciones:

- Variando ambos parámetros tenemos una función continua que crece proporcionalmente. Se va a utilizar para tomar las medidas del Power Saving (ver apartado 4.3).

- Variando sólo un parámetro tenemos una función constante. Se va a utilizar para el resto de las medidas

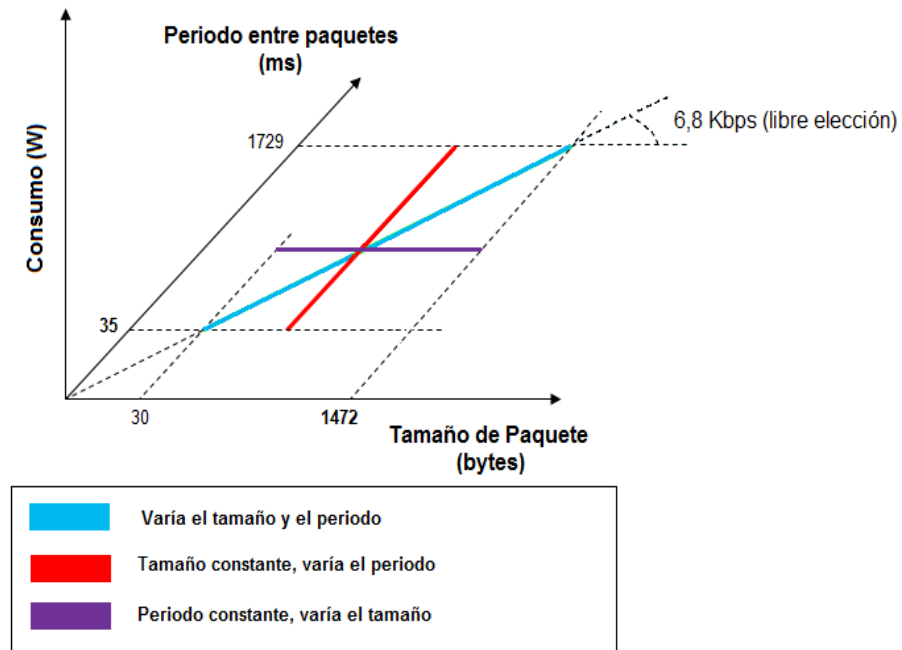


Figura 4.1: Experimentos

Entonces, tenemos que los experimentos son los siguientes:

1. Nivel de la batería. El consumo reflejado en el Nokia Energy Profiler depende del nivel de batería. Tomando la misma medida, con el nivel bajo consume menos que con la batería llena.
2. Power Saving. Cómo afecta la opción de ahorro de energía al consumo de batería.
3. Unicast. En una comunicación punto a punto, cómo afecta al consumo de la batería transmitir/recibir paquetes variando:
 - a) Tiempo entre paquetes.
 - b) Tamaño de paquete.
4. Multicast. En una comunicación punto a multipunto, cómo afecta al consumo de la batería transmitir/recibir paquetes variando:

- a) Tiempo entre paquetes.
 - b) Tamaño de paquete.
- 5. Symbian Vs Multicast. Se comparan las medidas anteriores entre los dos sistemas operativos.
 - a) Unicast
 - b) Multicast
- 6. RTS y CTS. Cómo afecta al consumo de batería cambiar estos parámetros del interfaz 802.11.

Los dispositivos a utilizar son el Nokia N810 Internet Tablet, y el N95.

En el N810 se ha tenido en cuenta que durante las medidas, el bluetooth esté desactivado, sin tener abierto el navegador, sin aplicaciones innecesarias abiertas y la luminosidad de la pantalla apagada (ver capítulo Problemas encontrados, Consumo de la batería cuando el display está apagado 5.3.1).

Las medidas se han tomado cerca del punto de acceso, aproximadamente 1 metro, porque no sabíamos si la distancia influiría. Por motivos prácticos, luego se ha realizado un experimento con la distancia y se vió que no afectaba demasiado al consumo de la batería.

Los valores iniciales para los experimentos son los siguientes:

VALORES DEL PUNTO DE ACCESO:

- Authentication Type: AUTO
- Basic Rate: DEFAULT
- Transmission Rate: AUTO
- CTS Protection Mode: DISABLE
- Frame Burst: DISABLE
- Beacon Interval: 100 ms
- DTIM Interval: 1
- Fragmentation Threshold: 2346
- RTS Threshold: 2347
- AP Isolation: OFF
- Secure Easy Setup: ENABLE

VALORES DE N810:

- Battery full
- Brightness of Display: Off

VALORES DEL PROGRAMA TTCP:

- Ordenador: `./ttcp -r -u -s -l -n -p5000`
- Maemo: `./ttcp -t -u -s -l -n -w -p5000 192.168.1.82`

VALOR DEL PROGRAMA NOKIA ENERGY PROFILER:

- Measurement Period: 1

4.2. Estudio del nivel de batería

Cuando se empezaron a tomar las medidas para el estudio asociado a la interfaz 802.11 del Nokia N810, el nivel de la batería no era el mismo en todas ellas, puesto que llevarlas a cabo conlleva un desgaste. Si tomábamos las medidas para ver cómo afecta al consumo para unas determinados valores, y la volviamos a repetir, el consumo reflejado en el Nokia Energy Profiler no era el mismo.

Se decidió realizar una medida sencilla para comprobar que la diferencia de consumo era a causa del nivel de batería. Este experimento consiste en tomar dos medidas con el Nokia Energy Profiler durante una conexión inalámbrica, una con la batería al máximo y otra al mínimo.

El resultado se refleja en la gráfica 4.2. Batería a nivel bajo (verde) frente a nivel alto (rojo), dando una diferencia aproximada de 0.12 W.

Como conclusión, tras ver cómo el nivel de la batería afecta al consumo, se decide tomar todas las medidas con la batería llena. Tras cada muestra, se recarga la batería.

4.3. Power Saving

En cuanto al Power Saving, el N810 presenta tres opciones: On Maximum, On Intermediate, Off. El modo On Maximum, corresponde a cuando el ahorro de energía está en funcionamiento, mientras que el modo Off, corresponde a cuando está apagado. Antes de realizar las medidas, se presupuso que la opción On Intermediate equivaldría a un estado intermedio entre activado y desactivado.

Además, es de esperar, que a medida que aumente el tiempo entre paquetes, el consumo sea cada vez mayor puesto que el terminal no tiene que estar activo para enviar paquetes continuamente.

Las medidas se han tomado de la siguiente manera:

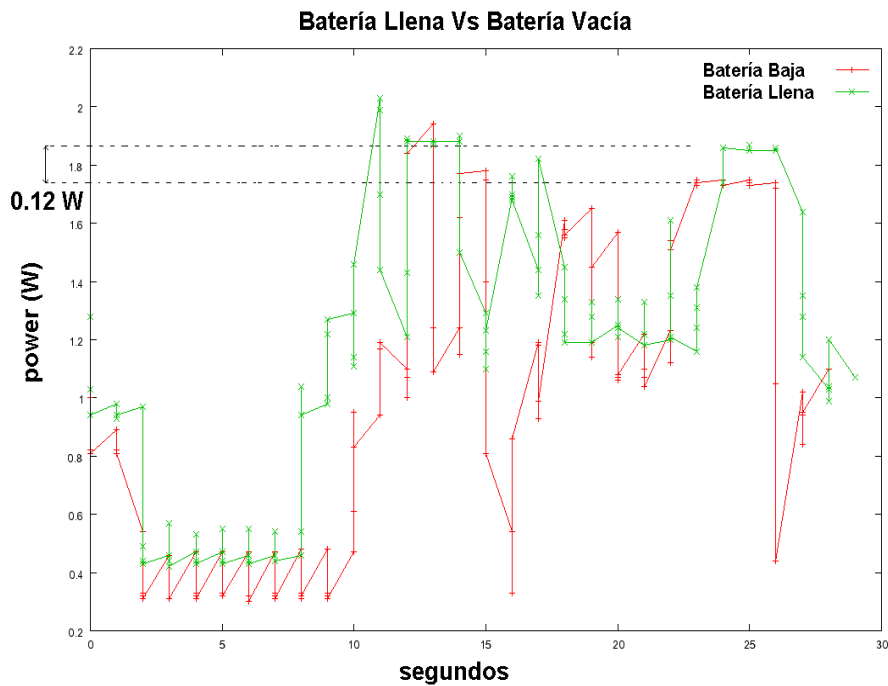


Figura 4.2: Nivel de la batería

500 KB en 600 seg (6,8 Kbit/s)

Power Saving: On Maximum, On Intermediate, Off

WLAN tx power: 10 mW, 100 mW

Transmisión

6 medidas:

- 1) 17067 paquetes de 30 bytes cada 35 ms
- 2) 1707 paquetes de 300 bytes cada 352 ms
- 3) 853 paquetes de 600 bytes cada 703 ms
- 4) 569 paquetes de 900 bytes cada 1056 ms
- 5) 427 paquetes de 1200 bytes cada 1408 ms
- 6) 348 paquetes de 1472 bytes cada 1729 ms

El resultado queda reflejado en las figuras, 4.3(a) para 10 mW, 4.3(b) para 100 mW. Ambas gráficas son muy parecidas, ya que el efecto de la potencia de transferencia en relación a la distancia del punto de acceso con el dispositivo móvil, no es relevante.

Sin embargo, el resultado de la opción Intermediate (verde) es inesperado, puesto que el consumo producido es similar al de la opción Off (azul). Cuando el

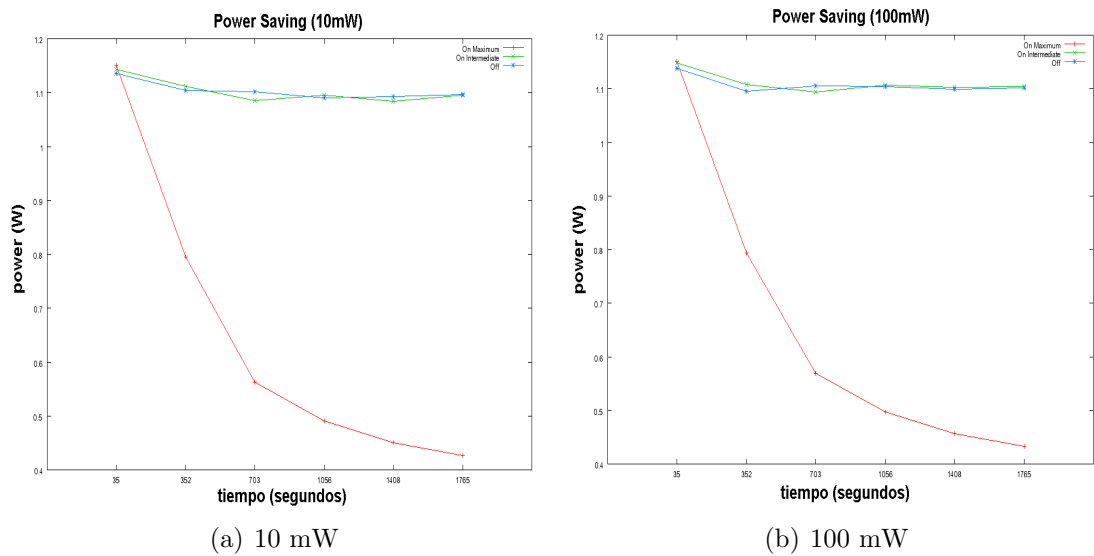


Figura 4.3: Power Saving

ahorro de energía esta activado, se produce una bajada del consumo.

Por otro lado, a medida que aumenta el tiempo entre paquetes, en el caso de On Maximum, la función decrece.

Por lo tanto:

- En cualquiera de las dos gráficas (4.3(a) o 4.3(b)) se puede observar la diferencia de potencia entre la opción *Maximum* y las otras dos. La diferencia entre ellas crece a medida que el tiempo entre paquetes aumenta. Lo cual también es lo que se esperaba puesto que el dispositivo N810 “duerme” más tiempo, y no tiene que estar enviando paquetes continuamente.
- Sin embargo, las opciones *Intermediate* y *Off* aparentemente son iguales. Para investigar el funcionamiento del ahorro de energía estando en *Intermediate*, se realizó un programa en el cual se realizaba la medida 6, se descansaba 5 minutos y se volvían a enviar paquetes:

```

1  #include <stdio.h>
2  #include <time.h>
3
4  int main(int argc, char *argv[]) {
5      if(argc != 2)
6          printf("Uso: ./funcionTtcp sleepSeconds\n");
7      else{

```

```

8      int tiempoEspera = atoi(argv[1]);
9      system("./ttcp -t -u -s -l1472 -n348 -w1729000
      -p5000 192.168.1.82");
10     time_t t = time(NULL);
11     printf("Comienza la espera: %u seconds\n",t);
12     sleep (tiempoEspera);
13     t = time(NULL);
14     printf("Termina la espera: %u seconds\n",t);
15     system("./ttcp -t -u -s -l1472 -n348 -w1729000
      -p5000 192.168.1.82");
16 }
17 return 0;
18 }

```

La figura 4.4 compara los 3 valores del Power Saving tras haber ejecutado el programa anterior. Como se puede observar, la función que equivale al valor Intermediate consume igual que cuando está apagado el ahorro de energía, hasta que detecta que no se envían paquetes y disminuye el consume hasta que coincide a cuando está activado. El intervalo de descanso que se aprecia en la figura, no es exactamente de 5 minutos, puesto que una vez que se dejan de enviar paquetes, no se produce el descenso de consumo inmediatamente, sino que realiza una espera de 4 segundos.

Cuando vuelve a enviar datos, el consumo aumenta hasta coincidir con la correspondiente a Off.

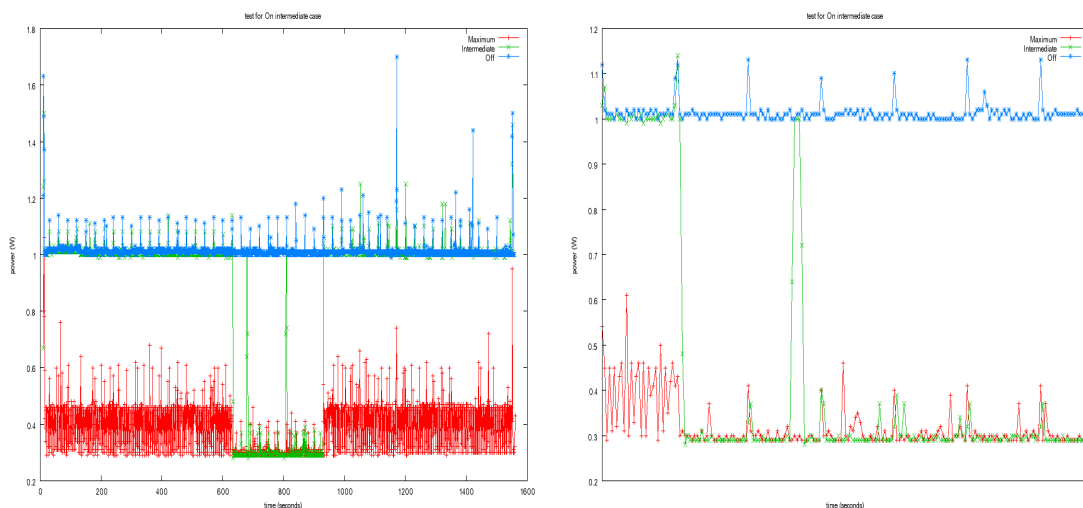


Figura 4.4: Power Saving en On Intermediate

4.4. Unicast

4.4.1. Variando tiempo entre paquetes

Lo que se espera con estas medidas, es comprobar si a medida que el tiempo es mayor entre paquetes enviados/recibidos, el consumo es menor.

Por lo tanto, se decidió hacer un experimento con los valores del siguiente cuadro, tanto para transmisión como recepción.

```
600 bytes
Power Saving: Off
WLAN tx power: 10 mW
Transmisión/Recepción
```

7 medidas:

- 1) 17067 paquetes de 600 bytes cada 35 ms
- 2) 1707 paquetes de 600 bytes cada 352 ms
- 3) 853 paquetes de 600 bytes cada 703 ms
- 4) 569 paquetes de 600 bytes cada 1056 ms
- 5) 427 paquetes de 600 bytes cada 1408 ms
- 6) 341 paquetes de 600 bytes cada 1765 ms
- Idle) sin paquetes

En las gráficas 4.5 se refleja un consumo que disminuye a medida que aumenta el tiempo entre paquetes, junto a una varianza muy amplia (identificada por las barras).

Si comparamos ambas gráficas (figura 4.6), el consumo en transmisión es similar al de recepción. La diferencia entre ambas es casi inexistente, aunque habría que tener en cuenta que hay una gran varianza en ambos casos.

En conclusión:

- En Unicast, las gráficas de transmisión y recepción son decrecientes con lo que se consume menos a medida que se incrementa el tiempo entre paquetes.
- El valor que toma Idle es el menor de todos (no se envían paquetes).
- Si se compara transmisión con recepción, tenemos que la diferencia entre ellas es despreciable (aproximadamente 4.5 mW en media). Esto no es lo

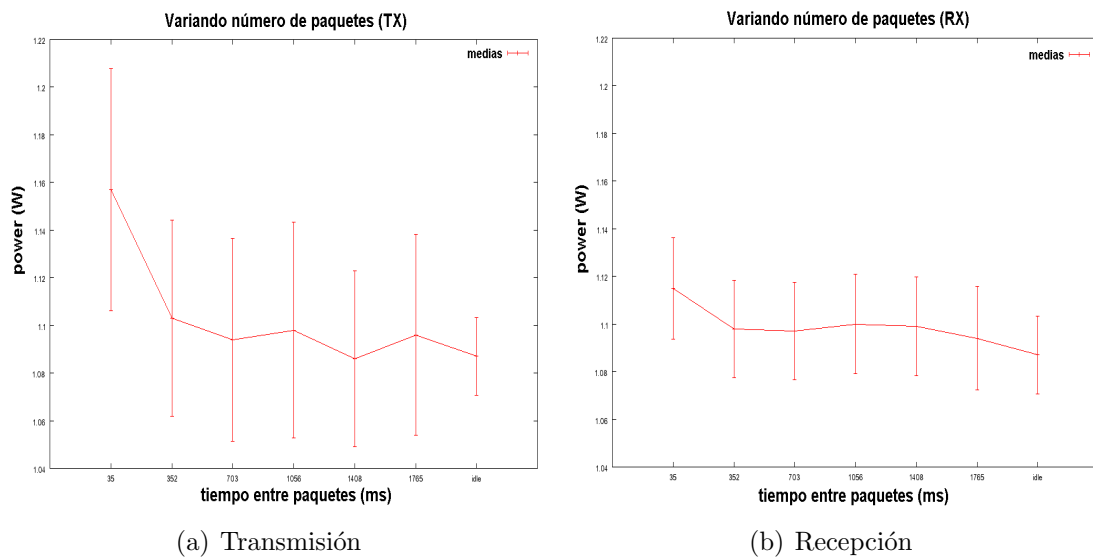


Figura 4.5: Unicast, Variando el tiempo entre paquetes

que se esperaba, lo normal es que en transmisión se consuma más, o que en recepción se consuma menos.

El resultado obtenido es el esperado tanto para transmisión como recepción. “En Unicast, el consumo disminuye a medida que el tiempo entre paquetes es mayor”.

4.4.2. Variando tamaño de paquetes

El resultado esperando de ir incrementado el tamaño de paquetes, es que el consumo sea aproximadamente constante; la media en todos casos debería de ser alrededor del mismo valor.

Se decidió realizar las siguientes medidas:

600 bytes
Power Saving: Off
WLAN tx power: 10 mW
Transmisión/Recepción

7 medidas:

- 1) 348 paquetes de 30 bytes cada 1729 ms
- 2) 348 paquetes de 300 bytes cada 1729 ms

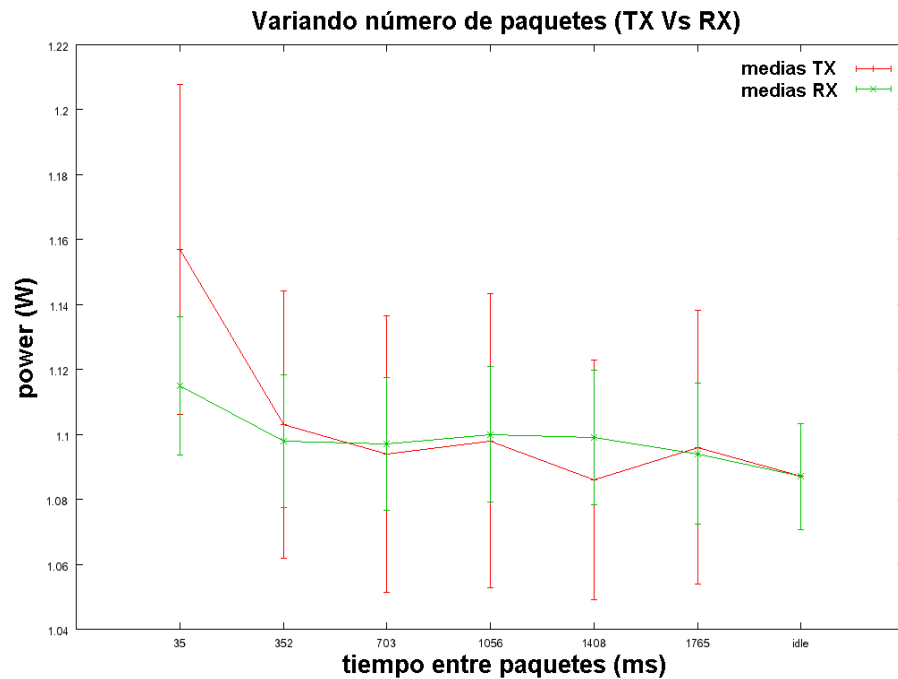


Figura 4.6: Unicast, Variando el tiempo entre paquetes - Transmisión Vs Recepción

- 3) 348 paquetes de 600 bytes cada 1729 ms
- 4) 348 paquetes de 900 bytes cada 1729 ms
- 5) 348 paquetes de 1200 bytes cada 1729 ms
- 6) 348 paquetes de 1472 bytes cada 1729 ms
- Idle) sin paquetes

Analizando las gráficas 4.7 tenemos que la de transmisión se mueve en valores entre 1.09 y 1.11, mientras que la de recepción, entre 1.09 y 1.10. A esos niveles de potencia, se puede decir que el consumo es constante.

En conclusión:

- En Unicast, tanto en transmisión como en recepción, las graficas son aproximadamente constantes en torno a un valor de 1.09 W.
- El valor que toma Idle es el menor de todos (no se envían paquetes).
- Si se compara transmisión con recepción, tenemos que la diferencia entre ellas es despreciable.

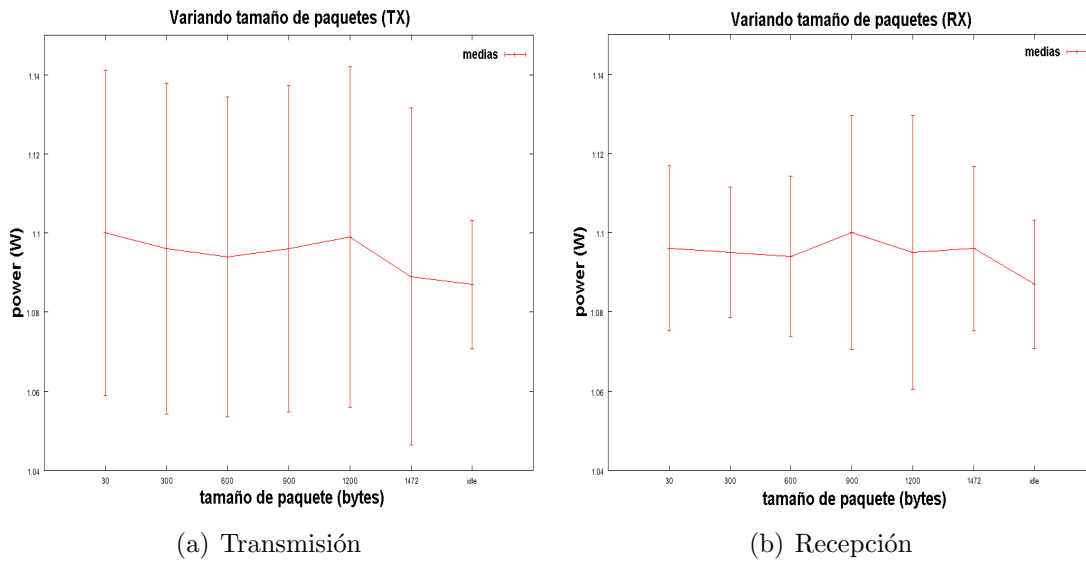


Figura 4.7: Unicast, Variando el tamaño de paquetes

Por lo tanto, ambas gráficas son igual de constantes (ver figura 4.8). “En Unicast, el tamaño del paquete no afecta al consumo de la batería del N810”.

4.5. Multicast

Para el entorno Multicast, ya explicado en el capítulo 2.1, se han cambiado los parámetros lanzados desde el programa `ttcp`:

- Maemo: `./ttcp -r -u -s -l -n -p5000 224.0.100.100`
- Ordenador: `./ttcp -t -u -s -l -n -w -p5000 224.0.100.100`

Lo que se espera con estas medidas es comprobar que ocurre lo mismo que en Unicast, y ver qué modo consume más batería.

NOTA: En ambos estudios, variando el número y tamaño de paquetes, se ha representado el valor del caso Idle e Idle estando asociado a un grupo multicast, obteniéndose un aumento en el consumo del último caso respecto al que no está asociado. Esta particularidad ocurre debido al funcionamiento del programa `ttcp` que se utiliza para enviar/recibir los paquetes. Para simular que está en modo Idle asociado, en la línea de comando de `ttcp` que se ejecuta en la consola desde el N810, se fija el tiempo (parámetro “w”) a un valor suficientemente alto para que no se corte la transmisión ni se envíen paquetes. De esta forma, según el código

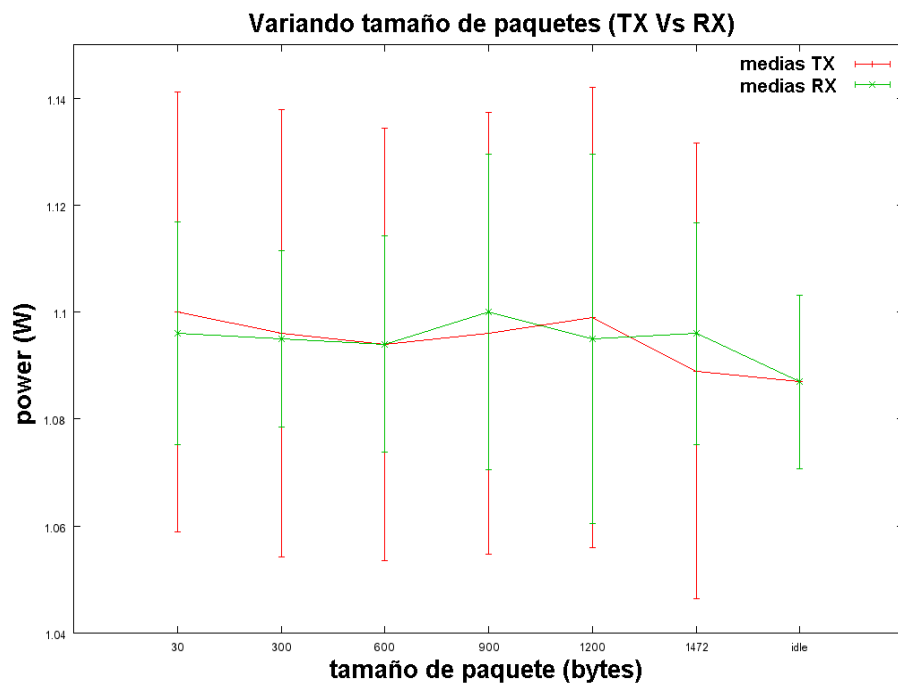


Figura 4.8: Unicast, Variando tamaño de paquetes - Transmisión Vs Recepción

del programa, hace una llamada a la función `select()` que consume una potencia extra de 0.2 W (para más información, ver la sección de Problemas encontrados, Consumo extra del programa Ttcp 5.3.3).

4.5.1. Variando el tiempo entre paquetes

Aunque el Power Saving se encuentre en modo Off, la potencia consumida cuando el tiempo entre paquetes es mayor, debería ser menor, ya que el N810 no está en continuo funcionamiento.

Al igual que en Unicast, las medidas tomadas son las siguientes:

600 bytes
 Power Saving: Off
 WLAN tx power: 10 mW
 Transmisión/Recepción

8 tests:

1) 17067 paquetes de 600 bytes cada 35 ms

- 2) 1707 paquetes de 600 bytes cada 352 ms
 - 3) 853 paquetes de 600 bytes cada 703 ms
 - 4) 569 paquetes de 600 bytes cada 1056 ms
 - 5) 427 paquetes de 600 bytes cada 1408 ms
 - 6) 341 paquetes de 600 bytes cada 1765 ms
- Idle - no assoc.) sin paquetes
Idle - assoc.) sin paquetes

La figura 4.9 compara el consumo entre Unicast y Multicast variando el tiempo entre paquetes. Las gráficas tienen en su eje de abscisas los valores del tiempo entre paquetes, y dos estados en los que no se envían paquetes, Idle, e Idle estando asociado.

Tanto en transmisión como en recepción, Unicast y Multicast tienen el mismo comportamiento, decreciente, y con una gran varianza en todas las medias. Además, los valores de Multicast son más altos que los de Unicast.

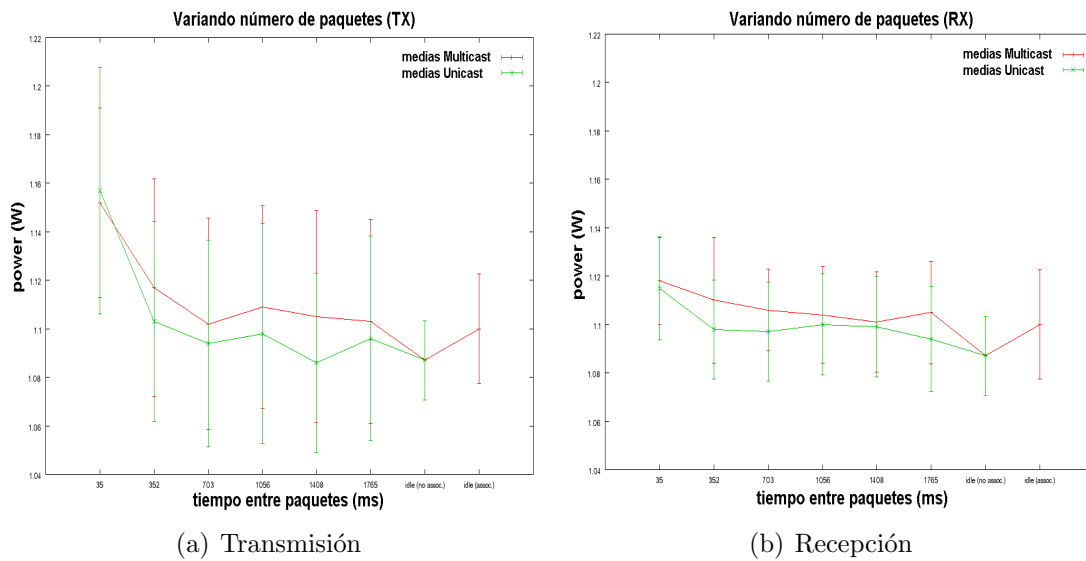


Figura 4.9: Multicast, Variando el tiempo entre paquetes

En conclusión:

- En Multicast, las gráficas de transmisión y recepción son decrecientes con lo que se consume menos a medida que el tamaño de paquete es mayor o el tiempo entre paquetes se incrementa.
- El valor que toma Idle sin estar asociado a un grupo de multidifusión, es el menor de todos (no se envían paquetes).

- Si se compara transmisión con recepción, tenemos que la diferencia entre ellas es despreciable (aproximadamente 7 mW en media).
- Si se compara con Unicast, tenemos que en Multicast, se consume más batería.

Por lo tanto, coincide con lo que se esperaba, “en Multicast, el consumo disminuye a medida que el tiempo entre paquetes es mayor”. Además, “el consumo de la batería es mayor en Multicast que en Unicast”.

4.5.2. Variando tamaño de paquetes

En este caso, se hace el estudio variando el tamaño de paquetes en lugar del número. En el caso Unicast, se ha llegado a la conclusión de que el tamaño no afecta al consumo de la batería, y como se espera, en Multicast debería ocurrir lo mismo.

600 bytes
 Power Saving: Off
 WLAN tx power: 10 mW
 Transmisión/Recepción

8 tests:

- 1) 348 paquetes de 30 bytes cada 1729 ms
- 2) 348 paquetes de 300 bytes cada 1729 ms
- 3) 348 paquetes de 600 bytes cada 1729 ms
- 4) 348 paquetes de 900 bytes cada 1729 ms
- 5) 348 paquetes de 1200 bytes cada 1729 ms
- 6) 348 paquetes de 1472 bytes cada 1729 ms
- Idle - no assoc.) sin paquetes
- Idle - assoc.) sin paquetes

Como se puede observar en la figura 4.10, tanto en transmisión como en recepción, las medias de Maemo y Symbian estan representadas aproximadamente por una función constante entorno a 1.103 W; ambas tienen el mismo comportamiento. Este valor es aproximado, debido a la gran varianza que presentan todas las medias.

En conclusión:

- En Multicast, tanto en transmisión como en recepción, las graficas son aproximadamente constantes en torno a un valor de 1.103 W.

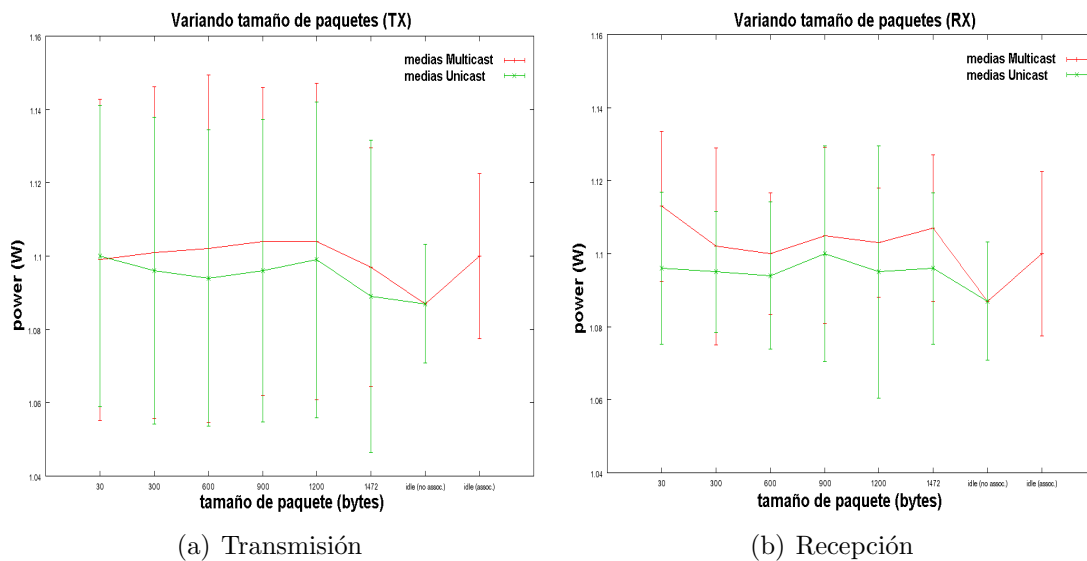


Figura 4.10: Multicast, Variando el tamaño de paquetes

- El valor que toma Idle sin estar asociado a un grupo de multidifusión, es el menor de todos (no se envían paquetes).
- Si se compara transmisión con recepción, tenemos que la diferencia entre ellas es despreciable.
- Si se compara con Unicast, tenemos que en Multicast, se consume más batería.

Por lo tanto, coincide con lo que se esperaba, “en Multicast, el tamaño de paquete no afecta al consumo de la batería del N810”. Además, “el consumo de la batería es mayor en Multicast que en Unicast”.

4.6. Symbian Vs Maemo

Como el proyecto se ha realizado en paralelo con el mismo estudio pero sobre otro dispositivo, Nokia N95, es interesante comparar el efecto que produce variar el tiempo entre paquetes y el tamaño de paquetes sobre dos dispositivos distintos que utilizan además, distinto sistema operativo, Maemo y Symbian.

Con esta comparación, se espera que las medidas tengan un comportamiento similar, y se espera saber cuál consume más.

Las medidas tomadas, siguen el mismo patrón que se ha explicado anteriormente para las medidas de Maemo.

4.6.1. Unicast

Variando el tiempo entre paquetes

En las gráficas de la figura 4.11 se puede apreciar una diferencia entre los valores de Maemo y de Symbian. En ambas ocasiones (transmisión/recepción) el consumo obtenido en Maemo es mayor que en Symbian, habiendo más diferencia en recepción (aproximadamente 0.02 W). De igual manera, hay una gran diferencia entre las varianzas, las resultantes de las medidas de Maemo, son mayores que las de Symbian. Sin embargo, el comportamiento que presentan ambas funciones, es similar.

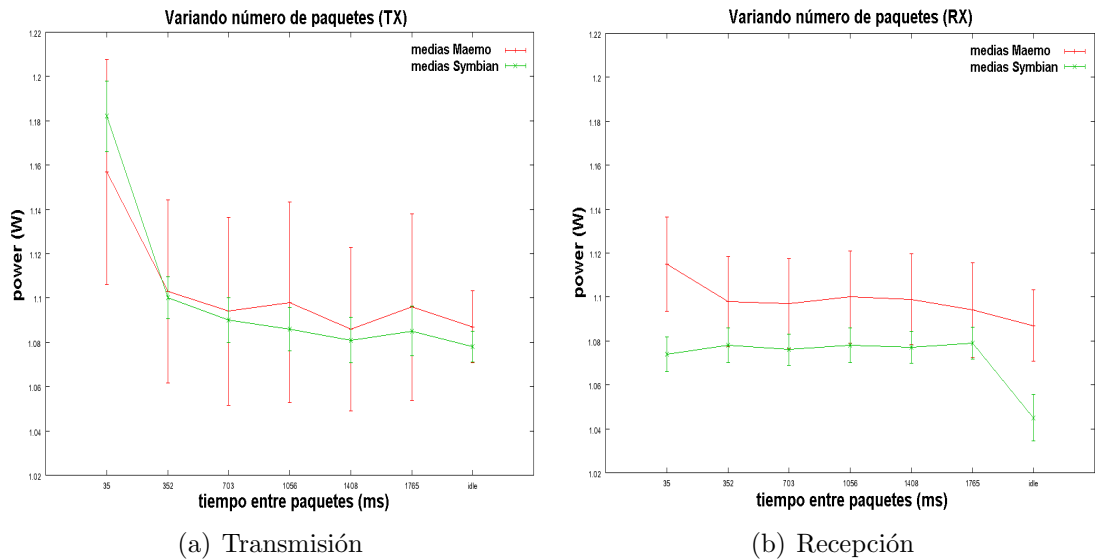


Figura 4.11: Symbian Vs Maemo: Unicast, Variando el tiempo entre paquetes

Como se esperaba en un principio, el comportamiento de ambas es igual, en transmisión, a medida que aumenta el tiempo entre paquetes, el consumo es menor. Mientras que en recepción, el tiempo entre paquetes es indiferente, el consumo es aproximadamente igual, entorno a 1.077 W para Symbian, y 1.1005 W en Maemo.

Además, cuando no se envía ningún paquete (estado Idle) la potencia consumida en Maemo, sigue siendo mayor que la de Symbian.

Por lo que en este caso, Maemo consume más batería que Symbian.

Variando tamaño de paquetes

Ambas gráficas (figura 4.12) tienen el mismo comportamiento, teniendo un valor más alto Maemo. Como ocurría cambiando el número de paquetes, la varianza de ambas funciones es distinta, siendo considerablemente mas alta la de Maemo.

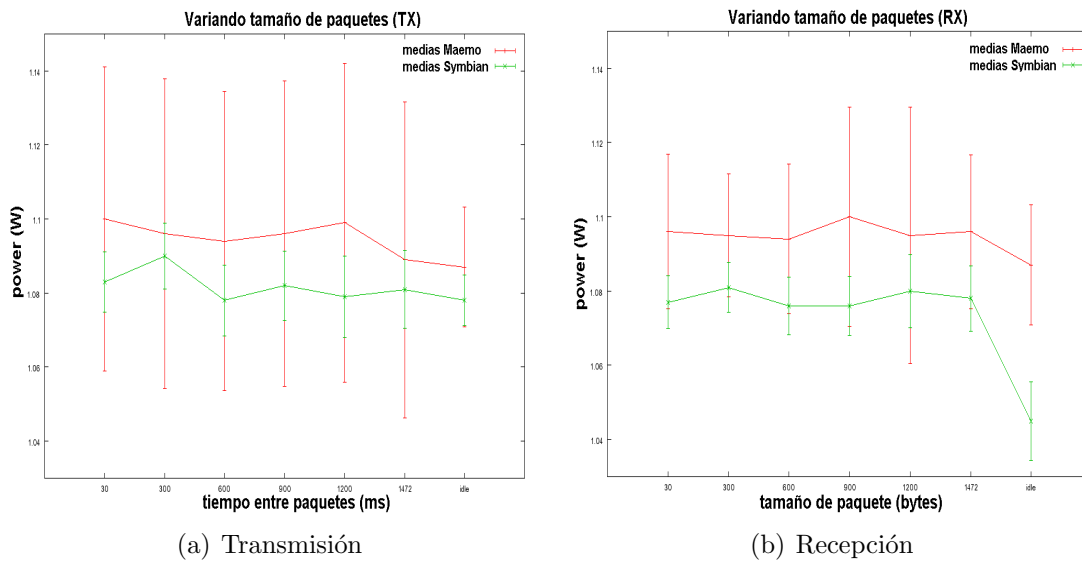


Figura 4.12: Symbian Vs Maemo: Unicast, Variando el tamaño de paquetes

En este caso, como ya se sabía, variar el tamaño de paquete no afecta al consumo de batería. Además, el consumo en Maemo es mayor que en Symbian, y aproximadamente constante, en transmisión en torno a 1.09 W para Maemo, 1.082 W para Symbian, y en recepción, 1.096 W para Maemo, 1.078 W para Symbian. Además, de esto se saca que en Maemo, el consumo es aproximadamente igual en transmisión que en recepción, mientras que en Symbian, el consumo es mayor en transmisión.

4.6.2. Multicast

Variando el tiempo entre paquetes

La figura 4.13 representa la comparación entre las medidas multicast en transmisión/recepción cuando varía el tiempo entre paquetes. Se puede apreciar la diferencia entre ambas funciones en ambos casos, tomando valores mas altos Maemo.

También hay una gran diferencia entre las varianzas, donde las de Maemo tambien son mayores. Las medidas toman valores mas pequeños a medida que aumenta el tiempo entre paquetes.

En las gráficas nos encontramos con dos posibles estados Idle, uno estando asociado a un grupo de multidifusión multicast, y otro sin estarlo. El primero de ellos da un valor menor que cuando está asociado en el caso de Maemo. En symbian, es aproximadamente igual, una diferencia de 0.001 W.

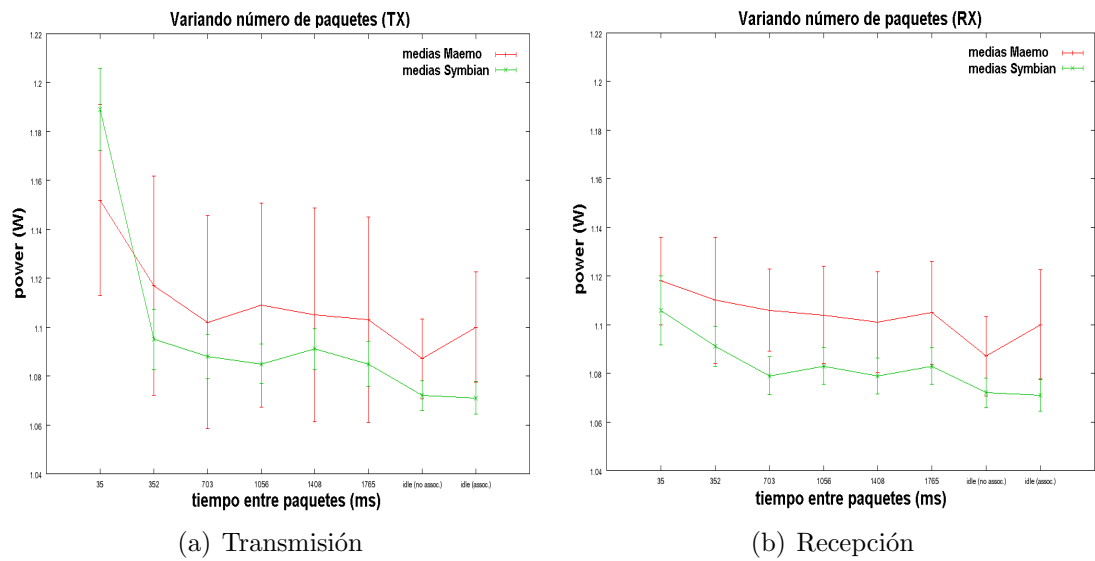


Figura 4.13: Symbian Vs Maemo: Multicast, Variando el tiempo entre paquetes

Por lo tanto, al igual que en Unicast, en transmisión, a medida que aumenta el tiempo entre paquetes, en Symbian y en Maemo, el consumo es menor. En recepción, el consumo disminuye en menor proporción que en transmisión. Como se esperaba, el comportamiento en Symbian es igual que en Maemo.

Solo hay una diferencia, el valor aumenta cuando el estado Idle está asociado a un grupo de multidifusión, que es debido al propio programa tcp que se utiliza para enviar/recbir paquetes(ver capítulo Problemas encontrados, Consumo extra del programa Ttcp 5.3.3).

Variando tamaño de paquetes

El último caso está reflejado en la figura 4.14. Como pasaba en Unicast, tenemos dos gráficas, donde Symbian y Maemo son similares, a excepción de que ésta ultima tiene un mayor consumo y como en todos los casos, las varianza en Maemo supera a la de Symbian.

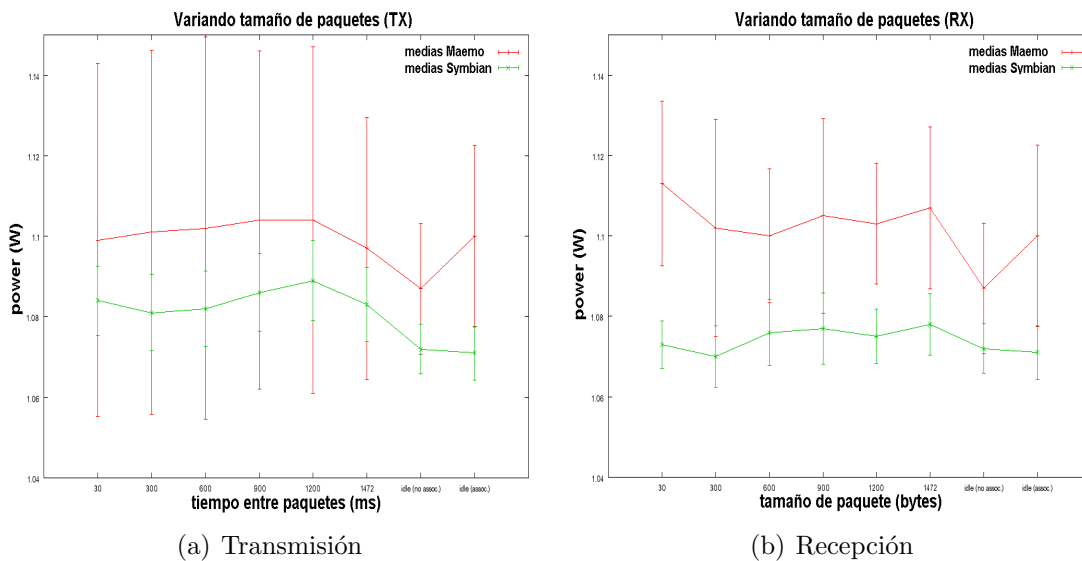


Figura 4.14: Symbian Vs Maemo: Multicast, Variando el tamaño de paquetes

Como se esperaba, tanto en transmisión como en recepción, el consumo es aproximadamente constante, en transmisión, 1.103 W para Maemo, 1.084 W para Symbian, y en recepción, 1.103 W para Maemo, 1.075 W para Symbian. Además, el consumo en Maemo es aproximadamente igual en ambos casos, mientras que en Symbian, es mayor en transmisión.

Por otro lado, al igual que sucede al variar el número de paquetes, se obtiene un valor elevado en Idle estando asociado a un grupo de multidifusión, debido al programa `ttcp` (ver capítulo Problemas encontrados, Consumo extra del programa `Ttcp` 5.3.3).

4.7. RTS y CTS

Conocido el funcionamiento de RTS y CTS (ver capítulo Tecnología de comunicación, Wi-Fi 2.1), se espera que el consumo de la batería estando estos parámetros activados sea menor.

Por lo tanto, se ha decidido hacer las siguientes medidas para RTS=29 y RTS=500:

```
500 KB
Power Saving: OFF
WLAN tx power: 10 mW
```

Transmisión

3 medidas:

- 1) 17067 packets of 30 bytes every 35 ms
- 2) 1707 packets of 300 bytes every 352 ms
- 6) 348 packets of 1472 bytes every 1729 ms

Se ha decidido utilizar el valor de 29 para forzar a enviar el RTS desde el principio ya que su valor es menor que el tamaño del primer paquete (30 bites). Como los resultados obtenidos no han sido significativos, se decide utilizar un valor mas grande, 500.

4.7.1. RTS=29 y CTS=AUTO

A la vista de los resultados, la diferencia entre estar activado o no, es despreciable. Ambas varianzas son muy elevadas, lo que implica un problema a la hora de analizarlas, ya que los datos pueden variar en un gran rango de valores.

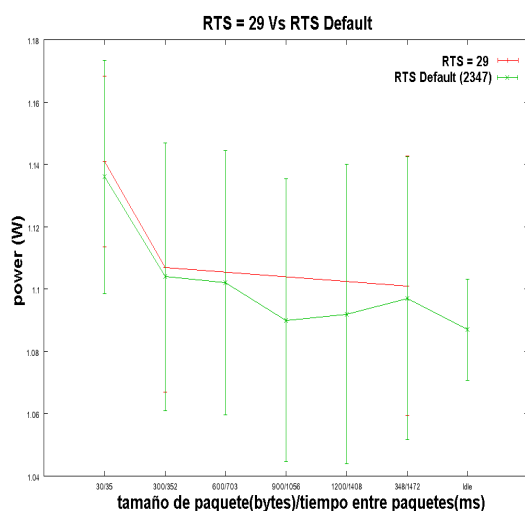


Figura 4.15: Efecto del RTS a 29 y CTS Auto

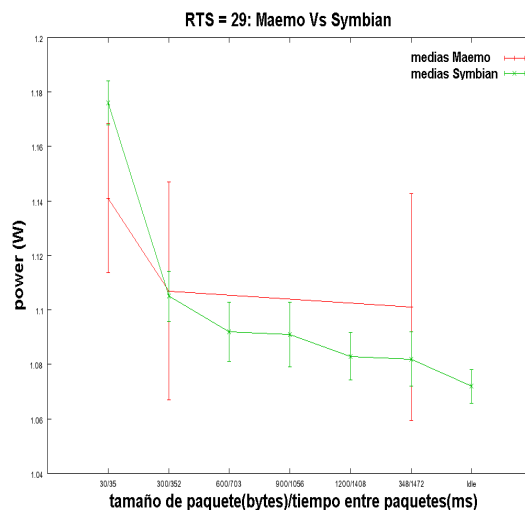


Figura 4.16: Symbian Vs Maemo: Efecto del RTS a 29 y CTS Auto

4.7.2. RTS=500 y CTS=AUTO

La interpretación de la gráfica 4.17(a) sigue siendo la misma que en el caso anterior. La diferencia entre ambas es despreciable.

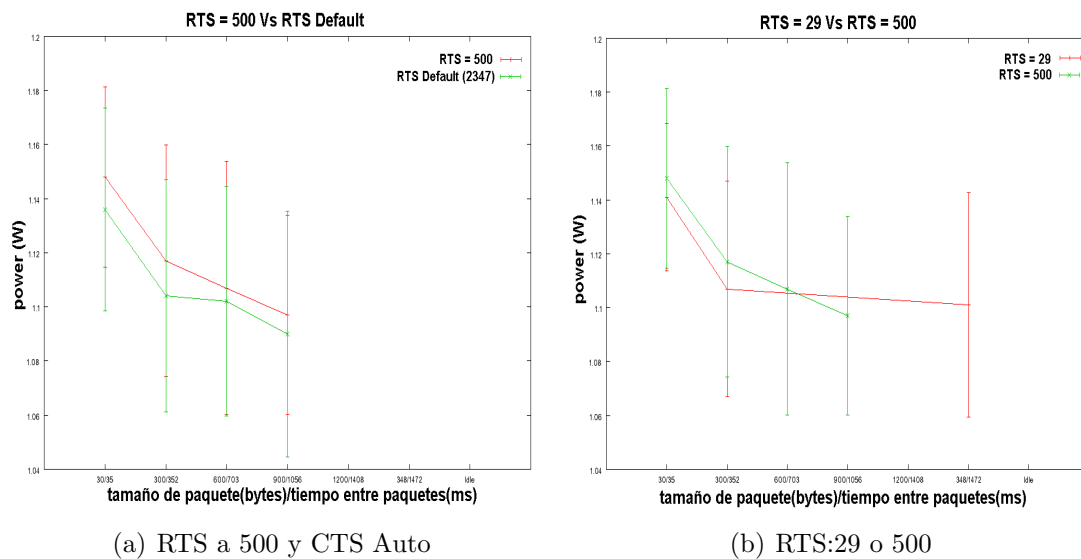


Figura 4.17: Efecto del RTS y CTS

El resultado es incongruente, por lo que hace pensar si realmente se envían RTS. Se intentó rastrear la red utilizando un sniffer (kismet¹), el cual detecta los beacons, pero por el momento no detectaba ningún RTS.

Por lo tanto, se ha decidido posponer como posible trabajo futuro y saber si realmente se envían o no los RTS, y el efecto que tiene.

4.8. Resultados

Antes de realizar los experimentos

Antes de realiza los experimentos partíamos con unas ideas de lo que podriamos obtener:

- Cuando el ahorro de energía está activado, el consumo de la batería debe ser menor que cuando está en la opción “Intermediate” y éste a su vez debe consumir menos que cuando está apagado. Cuando el Power Saving está activado, el N810 está durmiendo y sólo se despierta para transmitir (para saber más ver Tecnología de la Comunicación, Wi-Fi 2.1).
- Tanto en Unicast como en Multicast:

¹Programa de captura de las tramas de red que funciona con cualquier tarjeta inalámbrica que soporte el modo de monitorización raw, y puede rastrear tráfico 802.11b, 802.11a y 802.11g.

- A medida que aumenta el tiempo entre paquetes, el consumo es menor. El N810 no está continuamente transmitiendo paquetes porque el intervalo de tiempo es mayor.
 - Cuando varía el tamaño de paquete, el consumo de la batería debería ser aproximadamente igual en todo momento.
 - El consumo en transmisión debería ser mayor que en recepción.
- Dado el funcionamiento de RTS y CTS, cuando está activado se espera que el consumo sea menor.

Resultados no esperados

Tras realizar los experimentos, se han obtenido varios resultados no esperados:

- En las medidas de Power Saving, el consumo cuando está la opción de “On Intermediate” es similar a cuando no hay ahorro de energía.
- El consumo de tomar las medidas de Unicast y Multicast en recepción, es similar al de transmisión. No sabemos qué factor hace que el consumo en ambos modos sea similar. Lo único que podemos pensar es que es a causa del driver del dispositivo, y esto produce una baja eficiencia.
- El consumo estando en estado Idle asociado a un grupo multicast consume más que cuando no está. Este hecho es debido al código del programa Ttcp (para saber más, ver Problemas encontrados, Consumo extra del programa Ttcp 5.3.3).
- El efecto del RTS y CTS es despreciable, el consumo es similar estando activado o no.

Por otro lado nos ha llamado la atención la alta varianza obtenida en las medidas realizadas para el N810. Con los datos obtenidos de las medidas de los experimentos podemos pensar que es a causa del dispositivo, no de la tarjeta de red, ya que cuando no se envían paquetes (Idle) se sigue produciendo una gran varianza.

Capítulo 5

Historia del proyecto

Este proyecto surge a partir de un trabajo de investigación de la Cátedra Nokia del Departamento de Telemática de la Universidad Carlos III de Madrid. En este capítulo se explican las diferentes etapas del proyecto, el presupuesto y los distintos problemas encontrados en el camino.

5.1. Tareas

El estudio se ha llevado a cabo dentro del marco de la Cátedra de Nokia de la Universidad Carlos III de Madrid durante los meses de Marzo 2008 a Febrero de 2009. Se han llevado a cabo distintas fases de desarrollo, divididas en tareas, las cuales se explican a continuación.

5.1.1. Tarea 0, Configuración del escenario

El primer paso que se dió, es la configuración del escenario descrito en el capítulo 3.

Se han instalado los dos equipos con una distribución Linux limitada (uno para las medidas de Symbian, y otro para Maemo). Además, se han instalado las tarjetas de red (Ethernet y Wi-Fi), y se han creado unas cuentas locales para poder utilizar los equipos.

Por otro lado, se ha configurado en dispositivo Nokia N810 flasheandolo¹ e instalando los paquetes tal y como se muestra en el apéndice D.

¹Cambiarle el firmware interno.

5.1.2. Tarea 1, Estudio del dispositivo N810 y de la red Wi-Fi

El siguiente paso fué documentar los modos teóricos de consumo de energía en el estandar 802.11, así como los parámetros configurables en el punto de acceso y en el dispositivo.

5.1.3. Tarea 2, Prueba del escenario - Primeras medidas

Una vez que el escenario ya está configurado, incluyendo que el dispositivo ya tiene la versión del software requerida, con el programa de tcp y Nokia Energy Profiler funcionando, se realiza una primera medida básica para comprobar que todo está correcto.

Esta medida es la que se encuentra en el apartado 4.2. Sin embargo, cuando pasa un tiempo y la pantalla se apaga, el consumo que se muestra en el programa Nokia Energy Profiler es aproximadamente 0 W, problema que se soluciona como se muestra en el apartado 5.3.1.

5.1.4. Tarea 3, Medidas

Cuando todo está correcto, se empiezan las pruebas descritas en el capítulo 4. A la hora de llevarlas a cabo, siempre se ha seguido una metodología para que todo fuera más esquemático, y más facil de repetirlas, ver apéndice E.

5.2. Presupuesto

En esta parte se presenta el presupuesto asociado al desarrollo del presente proyecto. En primer lugar, se detallan los costes de personal en que se ha incurrido y, a continuación, se muestran los costes derivados del material empleado.

5.2.1. Costes de personal

Los costes de personal incluyen los honorarios del Ingeniero Técnico de Telecomunicación encargado del proyecto. Aunque el ritmo de trabajo no ha sido constante, se estima que se han invertido aproximadamente 5 horas al día durante los meses de Marzo de 2008 a Enero de 2009, excepto el mes de Agosto. Un total de 10 meses, que equivale a 1015 horas.

El baremo orientativo del COITT, donde se presentan los honorarios para un Ingeniero Técnico de Telecomunicaciones ya no son de carácter público, por lo que nos hemos basado en un sueldo actual para recién titulado de unos 21.000 €/año bruto. Contando 14 pagas, tenemos 1.500 €/mes, por lo que nos da un total de 6,25 €/hora bruto.

Concepto	Horas	Honorarios	Importe
Ingeniero Técnico de Telecomunicaciones	1015 horas	6,25 €/hora	6.343,75 €

Cuadro 5.1: Costes de personal

5.2.2. Costes del material

Para el desarrollo del proyecto, se han empleado los siguientes materiales:

- Ordenador personal de sobremesa con sistema operativo Linux, valorado aproximadamente en 600 €.
- Nokia N810 Internet Tablet, cuyo precio extraído de la tienda *online* de Nokia es de 450 €.
- Punto de acceso Linksys WRT54G, valorado aproximadamente en 60 €.

Concepto	Unidades	Precio unitario	Importe
Ordenador sobremesa	1	600€	600€
Nokia N810 internet Tablet	1	450€	450€
Linksys WRT54G	1	60€	60€
TOTAL			1.110€

Cuadro 5.2: Costes de material

5.2.3. Coste total

El presupuesto final es la suma de ambos costes, reflejado en la siguiente tabla:

Concepto	Importe
Costes de personal	6.343,75€
Costes de material	1.110€
TOTAL	7.453,75€

Cuadro 5.3: Coste total

5.3. Problemas encontrados

5.3.1. Consumo de la batería cuando el display está apagado

Se ha encontrado un error en la aplicación Nokia Energy Profiler una vez que se apaga el display y se queda la pantalla en negro. El error que se produce es que el descenso de la potencia consumida en ese instante es de 0.01 W (ver imagen 5.1), y no debería consumir tan poco habiendo otros procesos en funcionamiento.

El sistema operativo instalado en el dispositivo es maemo linux based OS2008 cuya versión es 0.2008.12-14. Por otro lado la versión instalada del Nokia Energy Profiler es 0.5.

Dentro de la configuración del N810 tenemos dos valores que influyen en el display y que se pueden cambiar:

- Display Settings, Switch off Display: 5 minutes, indica cuándo se apaga la pantalla.
- Brightness period, indica el periodo de tiempo que debe transcurrir hasta que el brillo de la pantalla se queda apagada.

Sin embargo para los experimentos realizados en este proyecto no es suficiente, ya que cada medida tarda aproximadamente 10 minutos, y según la configuración, permite como máximo 5 minutos, por lo que se apagaría la pantalla a mitad de ésta.

Como ejemplo tenemos una captura del Nokia Energy Profiler (figura 5.1), donde se puede apreciar que a los 30 segundos se produce un descenso de la potencia, que coincide con los instantes en el que el display está apagado. Se ha

medido el consumo que produce tener conectado el bluetooth y se ha configurado la pantalla para que se apagara a los 30 segundos y comprobar así con el programa el descenso de potencia.

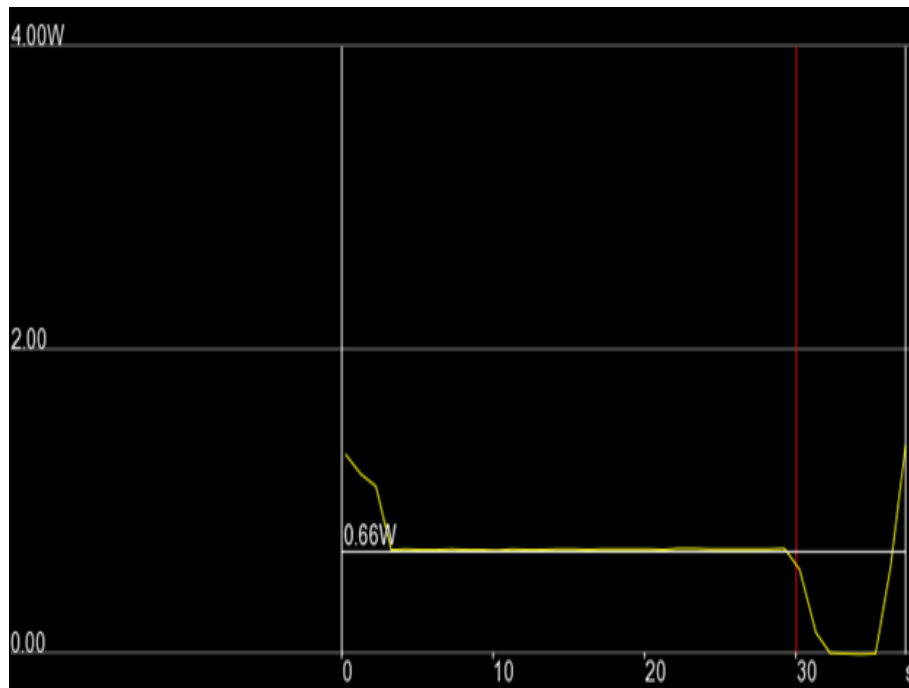


Figura 5.1: Error Nokia Energy Profiler

Reproducción del error

Cuando se han tomado los experimentos, nos hemos dado cuenta de que el error sólo se producía cuando el Power Saving tomaba el valor de On Maximum.

La figura 5.2 compara el consumo entre dos medidas tomadas mientras se conecta a la red inalámbrica del escenario (ver Capítulo Escenario 3), una con el Power Saving On Maximum (verde) y la otra cuando está apagado (roja). Como se puede observar, en el momento en que se apaga el display, se produce un descenso en el consumo, siendo aproximadamente 0 W en el caso de On Maximum.

Tomando como ejemplo la figura 5.1, lo que hay que hacer para reproducir el error es lo siguiente:

1. Poner en marcha en Energy Profiler.
2. Arrancar una aplicación, por ejemplo el bluetooth.

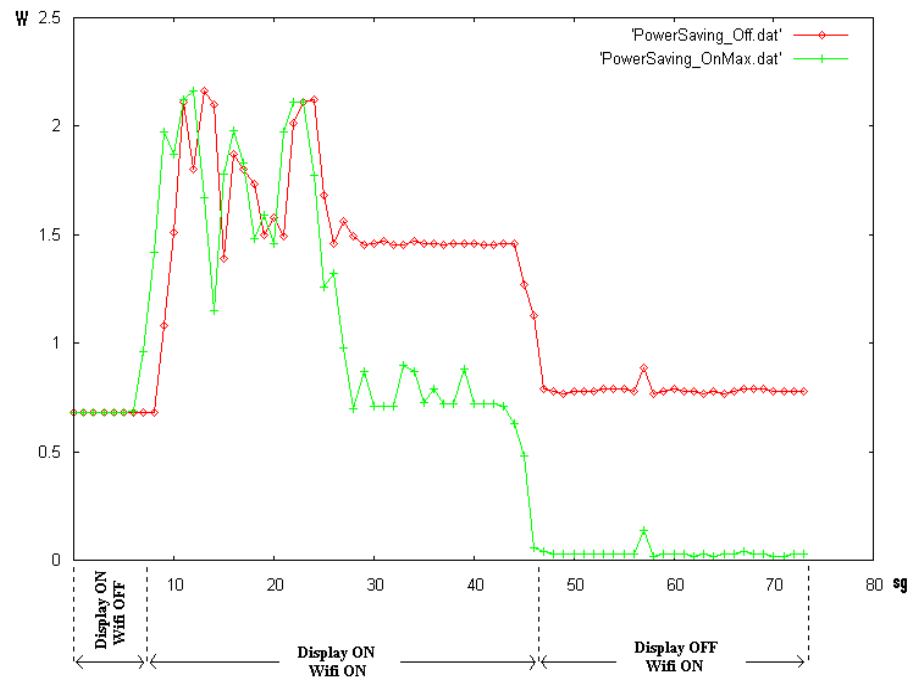


Figura 5.2: Error Nokia Energy Profiler-conexión a la red cambiando el valor de Power Saving

3. Dejar de interactuar con el dispositivo para que la pantalla se apague (en el ejemplo tras 30 segundos).
4. Dejar unos segundos la pantalla apagada.
5. Encender el display y detener el Energy Profiler.

De esta forma se puede apreciar el cambio en la potencia consumida, y como vuelve a aumentar cuando se enciende el display.

Solución

Hay dos posibles soluciones:

1. Display Settings, Switch off Display: 5 minutos. Tarda 5 minutos en apagarse el display. Si los experimentos que se vayan a medir duran más, esta solución no nos vale, habría que elegir la solución 2.
2. Hay que utilizar una aplicación llamada “gconf-editor” de la siguiente página e instalarlo con “Application Manager”:

<http://maemo.org/downloads/product/gconf-editor/>

Se trata de una aplicación para el entorno GNOME, y sirve para editar las preferencias de GConf, por lo tanto, útil para el sistema operativo Maemo, ya que sus pilares están formados por estos componentes.

A continuación, se abre y se selecciona el siguiente directorio:

`“/system/osso/dsm/display/”`

y se cambia el valor de “display_blank_timeout” a un valor suficientemente grande para que no se quede en off, por ejemplo a 86400 (el valor es en segundos, tratándose así de un día).

5.3.2. Compilación de código C en el entorno Maemo

Cuando se ejecuta un programa en el N810 y no ha sido compilado en el entorno Maemo, aparece el siguiente mensaje:

```
./HelloWorld: line 1: syntax error: '(' unexpected
```

En este caso, se trata de un código en C llamado “HelloWorld”.

```
1  \#include <studio.h>
2
3  int main(){
4      printf('';Hello World!\n'');
5      return 0;
6  }
```

Reproduccion del error

Para reproducirlo, lo único que hay que hacer es ejecutar el binario en la consola del N810:

1. Se guarda el binario en la memoria del N810.
2. Se abre la consola Xterm del N810.
3. Se cambia el path a donde se encuentre el binario
4. Se dan permisos de ejecución:

```
/media/mmc2 $ cp HelloWorld /tmp
/media/mmc2 $ cd /tmp
/tmp $ chmod 775 HelloWorld
/tmp $ ls -l | grep HelloWorld
-rwxrwxr-x 1 user  users 14105 May 15 13:06 HelloWorld
```

5. Se ejecuta el binario

```
./HelloWorld
```

Solución

Para solucionarlo, el binario debe ser compatible con el Sistema Operativo Maemo y por tanto debe de estar compilado en ese entorno. Para ello hay que seguir los pasos que se describen en el apéndice A, o bien consultar la página web ([1]).

5.3.3. Consumo extra del programa Ttcp

Cuando se comparan los experimentos con los de Symbian (ver Medidas, Symbian Vs Maemo 4.6), en las cuales en Symbian el consumo es similar en el caso de Idle estando o no asociado a un grupo multicast, mientras que en Maemo, se consume más cuando está asociado, hace pensar que el problema puede estar en el programa Ttcp. Por esta razón, tras mirar el código en profundidad, la parte orientada a enviar los paquetes (función Nwrite) realiza un select, dejando el programa en funcionamiento durante el período de tiempo que se le indique. De esta forma, consume una potencia extra de 0.02 W como se puede ver en la gráfica 5.3.

La parte del código es:

```
1  /*
2  /*
3  *      N W R I T E
4  */
```

```

5  int
6  Nwrite(int fd, void *buf, int count)
7  {
8      register int cnt;
9      if (udp) {
10 again:
11     cnt = sendto(fd, buf, count, 0, res->ai_addr, res->
12         ai_addrlen);
13     numCalls++;
14     if (cnt<0 && errno == ENOBUFS) {
15         delay(18000);
16         errno = 0;
17         goto again;
18     }
19     } else {
20     cnt = write(fd, buf, count);
21     numCalls++;
22     }
23     if (wrwait)
24         delay(wrwait);
25     return(cnt);
26 }
27
28 void
29 delay(int us)
30 {
31     struct timeval tv;
32
33     tv.tv_sec = 0;
34     tv.tv_usec = us;
35     (void)select(1, NULL, NULL, NULL, &tv);
36 }

```

Reproducción del error

Solamente con la versión 1.12, año 2002, que incluye la opción de tiempo entre paquetes (“w”), cuando la comunicación es Multicast y las medidas son en estado idle, ocurre el aumento del consumo.

Segun el código `ttcp.c`, al tener la opción “w” guarda el valor del tiempo en una variable llamada **wrwait**. De esta forma, cuando entra en el método **nwrite**, hace el **if** y llama al método **delay**. En él se crea la estructura `timeval` que indica el tiempo que queremos esperar como máximo.

Luego hace una llamada al **select**², el cual va a retornar cuando pase el tiempo especificado en el timeval.

Este proceso hace que haya un consumo extra de 0.02 W, reflejado en la gráfica 5.3.

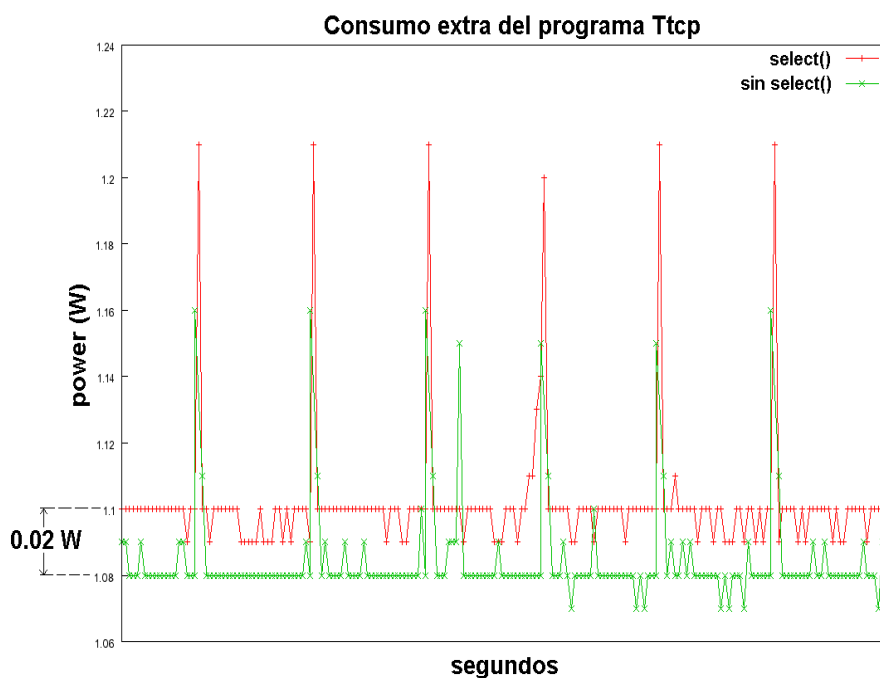


Figura 5.3: Ttcp - Consumo extra

Para poder reproducirlo hay que hacer lo siguiente:

1. Introducir la línea de comandos en el ordenador para la opción Multicast.

```
./ ttcp -r -u -s -l -n -p5000 224.0.100.100
```

2. Lanzar el programa Nokia Energy Profiler.

3. Introducir la línea de comandos en el N810 para la opción Multicast.

```
./ttcp -t -u -s -n -l -w -p5000 224.0.100.100
```

²Función que nos permite monitorizar un conjunto de descriptores de sockets y nos avisa cuáles tienen datos para leer, cuáles están listos para escribir, y cuáles produjeron excepciones.

El error se reproduce cuando se toman las medidas en el estado Idle, por lo que el tiempo debe ser aproximadamente igual al que se tarda en enviar/-recibir todos los paquetes (va con la opción “w”).

4. Terminar la comunicación y cerrar el programa.

Solución

Para solucionarlo habría que cambiar el código del programa y evitar esa llamada a la función select, con lo que habría que simular el caso en que está asociado a un grupo Multicast de otra manera. En este proyecto no se ha llegado a averiguar cuál sería la forma más eficiente de realizar este cambio. Sólomente se buscó el problema realizando un código con una llamada a la función select de un tiempo de espera grande que simulara el caso de estar asociado, y uno sin llamada que equivale al estado idle sin estar asociado, para ver la diferencia de consumo (ver gráfica 5.3).

Código para la simulación de estar asociado a un grupo multicast:

```
1  #include <stdio.h>
2  #include <sys/time.h>
3  #include <sys/types.h>
4  #include <unistd.h>
5
6  int main(int argc, char *argv[]) {
7      if(argc != 2)
8          printf("Uso: ./goSleep microseconds\n");
9      else{
10         int s = atoi(argv[1]);
11         struct timeval tv;
12
13         tv.tv_sec = 0;
14         tv.tv_usec = s;
15
16         select(1, NULL, NULL, NULL, &tv);
17     }
18     return 0;
19 }
20 }
```


Capítulo 6

Conclusiones y trabajos futuros

6.1. Conclusiones

Como conclusión de los experimentos realizados para el estudio del consumo asociado a la interfaz 802.11 del Nokia N810 Internet Tablet (Capítulo Medidas 4), tenemos que:

1. Efecto del Power Saving y de la potencia de transmisión en el consumo de potencia:

- La configuración del N810 permite que el Power Saving pueda tomar tres posibles valores, On Maximum (ahorro de energía activado), On Intermediate (ahorro de energía intermedio) y Off (ahorro de energía desactivado).

Con el experimento se obtiene que:

- Cuando está en On maximum el consumo es menor que el tomado por los otros dos valores.
- El consumo de potencia cuando está en On Intermediate es similar al consumo cuando está apagado.

- Estando a 1 metro de distancia con el punto de acceso, la diferencia de consumo de potencia de transmisión a 10 mW y a 100 mW es despreciable.

2. Efecto de una comunicación punto a punto y punto a multipunto en el consumo de potencia entre el N810 y el ordenador:

- A medida que aumenta el tiempo entre paquetes, disminuye el consumo.

- Si variamos el tamaño de paquete, no tiene influencia en la potencia consumida.
 - El consumo en transmisión es similar al de recepción
3. Se han podido comparar las medidas con las obtenidas en el sistema operativo Symbian dando como resultado que el consumo es menor que en Maemo.
 4. Por último, gracias al estudio de los parámetros configurables en el estándar IEEE 802.11, se han modificado el RTS y el CTS para ver el efecto que producía. Sin embargo, estas medidas han dado un resultado incongruente puesto que la diferencia estando activado o no es despreciable.

En definitiva, se han alcanzado todos los objetivos que estaban planteados al comenzar la realización del proyecto (ver Objetivos, 1.2) y se han planteado posibles líneas de trabajo futuro para continuar con la investigación.

6.2. Trabajos futuros

En esta sección proponemos una serie de trabajos futuros que se pueden llevar a cabo para la investigación sobre el consumo asociado a la interfaz 802.11 del Nokia N810 Internet Tablet:

- **Configuraciones del punto de acceso**

Además de continuar con las pruebas ya iniciadas del RTS/CTS, existen más configuraciones en el punto de acceso Linksys WRT54G, como cambiar los valores del Beacon, DTIM,... para saber cuál es la configuración óptima.

- **Roaming**

Roaming, capacidad de un dispositivo para moverse de una zona de cobertura a otra, desplazándose e ir registrándose en diferentes puntos de acceso. Asegura que el dispositivo inalámbrico se mantiene conectado a la red sin perder conexión.

Algo habitual, es que tratándose de un dispositivo móvil, el usuario cambiará de un punto de acceso a otro, por lo que es bastante obvio hacer un estudio para saber cómo afecta a la batería.

- **Envío/Recepción entre el N95 y el N810 Internet Tablet**

Las medidas se han tomado en paralelo con las realizadas por el N95. Teniendo ambos estudios, sería interesante realizarlas entre ambos dispositivos en vez de una comunicación móvil-Pc.

- **Estudio con el Nokia N900**

Como ya se ha mencionado en el apartado 2.5.1, la evolución del N810 es el N900, que en este caso sí se trata de un teléfono móvil, además de que incorpora la nueva versión del sistema operativo, Maemo 5. Con esto, se trata de realizar un nuevo estudio de batería en un teléfono y con la evolución del sistema operativo.

Apéndice A

Maemo SDK VMware

Introducción

Este apéndice presenta un ejemplo de compilación y ejecución de un fichero en C (HelloWorld.c) para el entorno Maemo.

Los pasos que hay que dar son los siguientes:

1. Obtener el VMWare¹ de Maemo e instalarlo en el ordenador.

La figura A.1 muestra el escritorio del SDK VMWare.

2. Abrimos el VMWare y nos vamos al Scratchbox Environment (suele aparecer en el Escritorio). Se trata de una maquina virtual que proporciona un ambiente de ejecución similar al de maemo utilizando dos ventanas: *Scratchbox*² y *Xephyr*. Para compilar y ejecutar solo nos interesa la primera, puesto que la segunda simula el interfaz gráfico del N700/800/810 (ver figura A.2).
3. A continuación, hay que realizar dos pasos para cambiar el tipo de target del scratchbox, ya que por defecto viene CHINOOK_X86 (i386) y nos interesa CHINOOK_ARMEL (arm):
 - Sin arrancar la simulación del N700/800/810 hay que llamar a *sb-menu* (ver figura A.3).
 - Y cambiar el target en la opción **Select** (ver figura A.4).

¹Es un programa que permite ejecutar (simular) varios ordenadores (sistemas operativos) dentro de un mismo hardware de manera simultánea, permitiendo así el mayor aprovechamiento de recursos

²Conjunto de herramientas de compilación que permiten el desarrollo de aplicaciones Linux

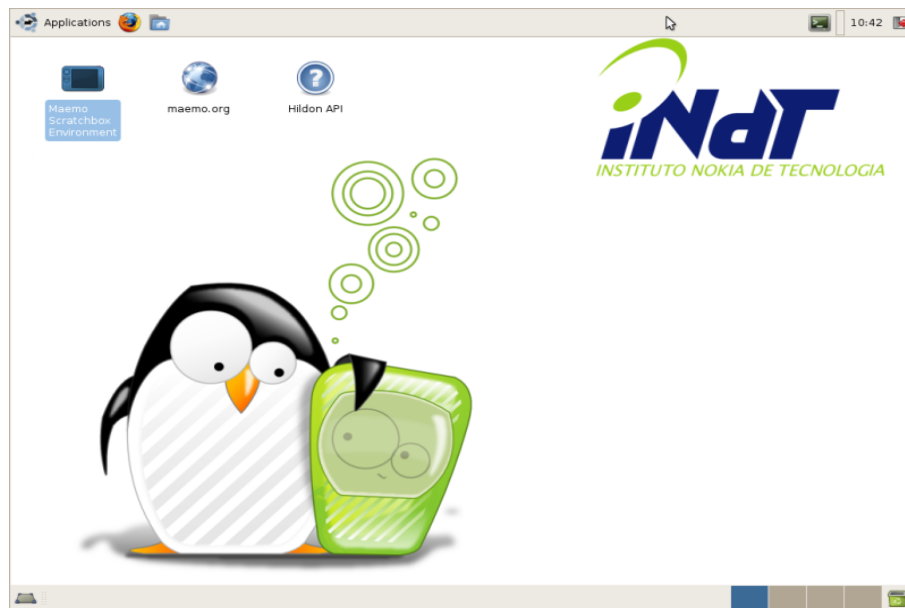


Figura A.1: Maemo SDK VMWare Appliance v0.6

4. Se crea un archivo Makefile para compilar el fichero HelloWorld.c
HelloWorld.c :

```

1  \#include <studio.h>
2
3  int main(){
4      printf('';Hello World!\n'');
5      return 0;
6  }
```

Makefile:

```

1  CC=gcc
2  BINARY=HelloWorld
3  LIBS=hildon-1 gtk+-2.0
4
5  all: HelloWorld
6      @echo ' ' ' '
7      @echo ''to run the compiled binary:''
8      @echo ''$$ run-standalone.sh ./${(BINARY)}''
9
```

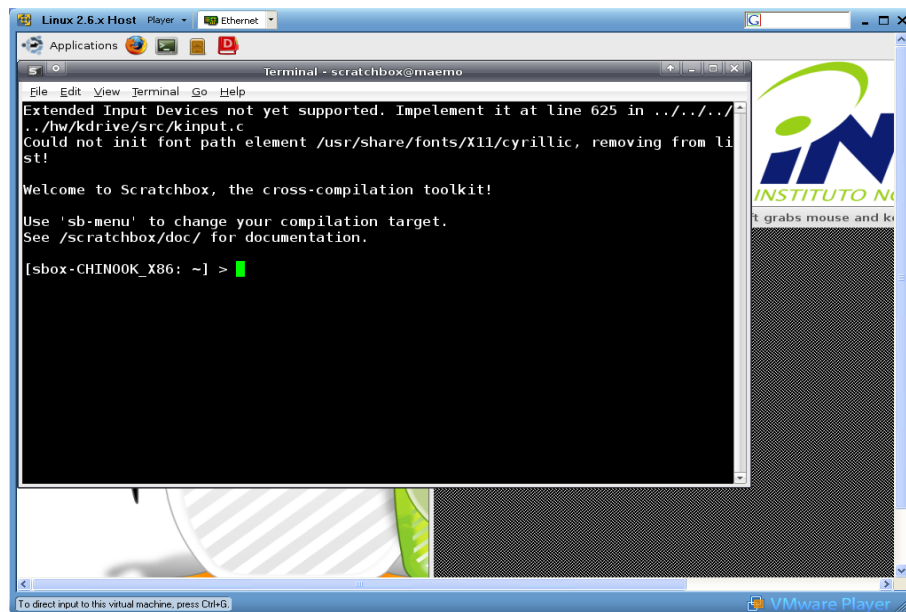


Figura A.2: Scratchbox

```

10 HelloWorld: HelloWorld.o
11     gcc 'pkg-config --libs $(LIBS)' HelloWorld.o -o
12         $(BINARY)
13
14 HelloWorld.o: HelloWorld.c
15     gcc 'pkg-config --cflags $(LIBS)' $(BINARY).c -c
16
17 clean:
18     rm -f *.o $(BINARY)

```

5. En el Scrathcbox se introduce el comando make que genera el binario del fichero HelloWorld, y utilizando el script *run-standalone.h* se ejecuta el binario creado (ver figuras A.5 y A.6).

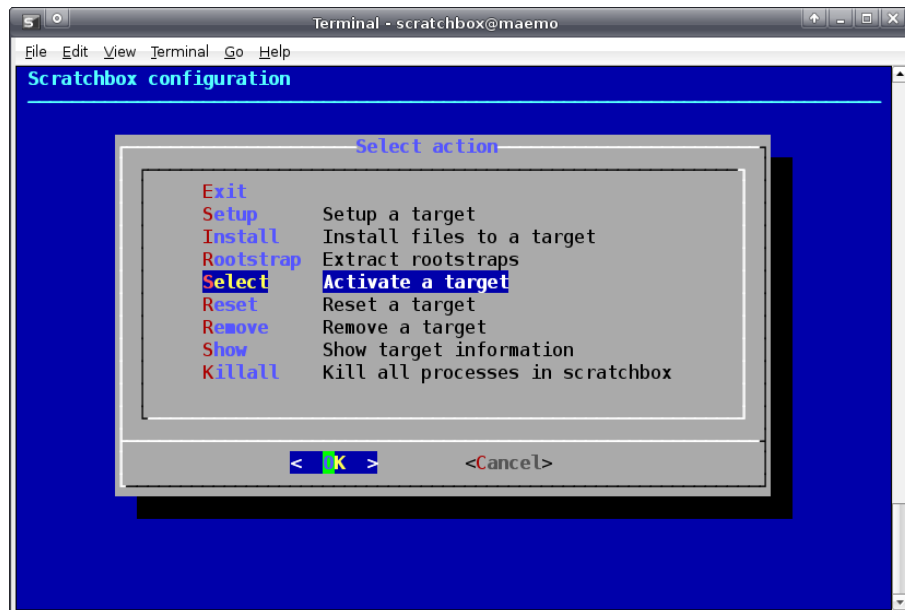


Figura A.3: Resultado tras ejecutar *sb-menu* en el scratchbox

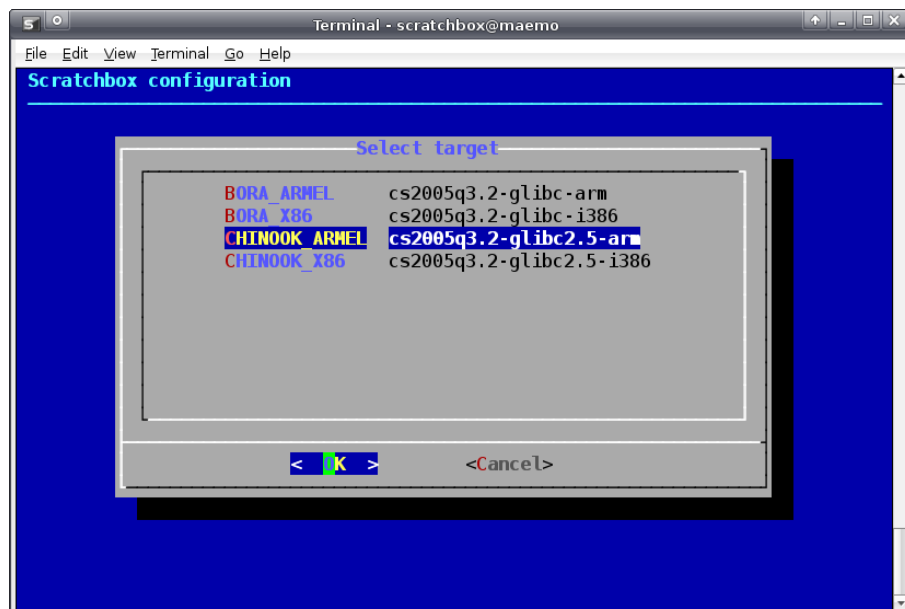
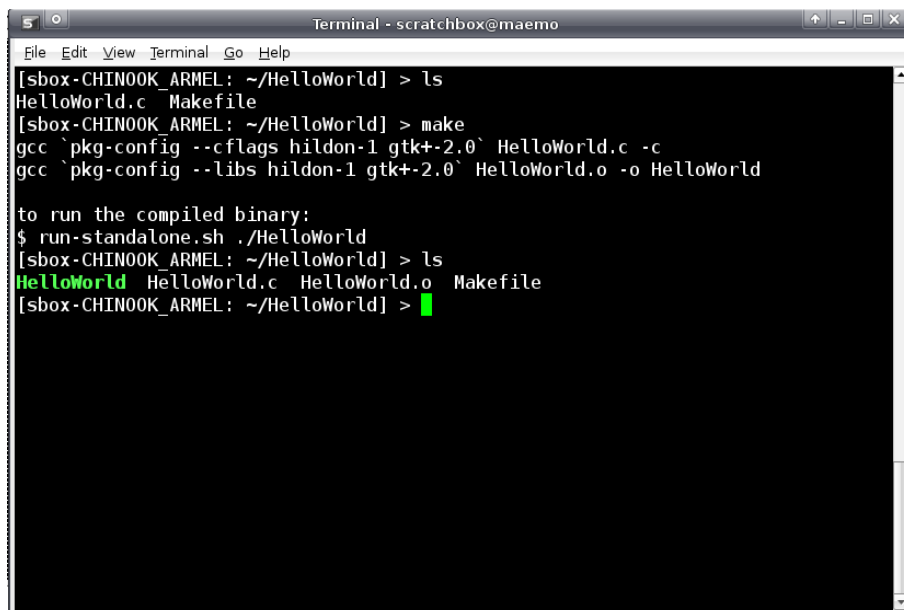


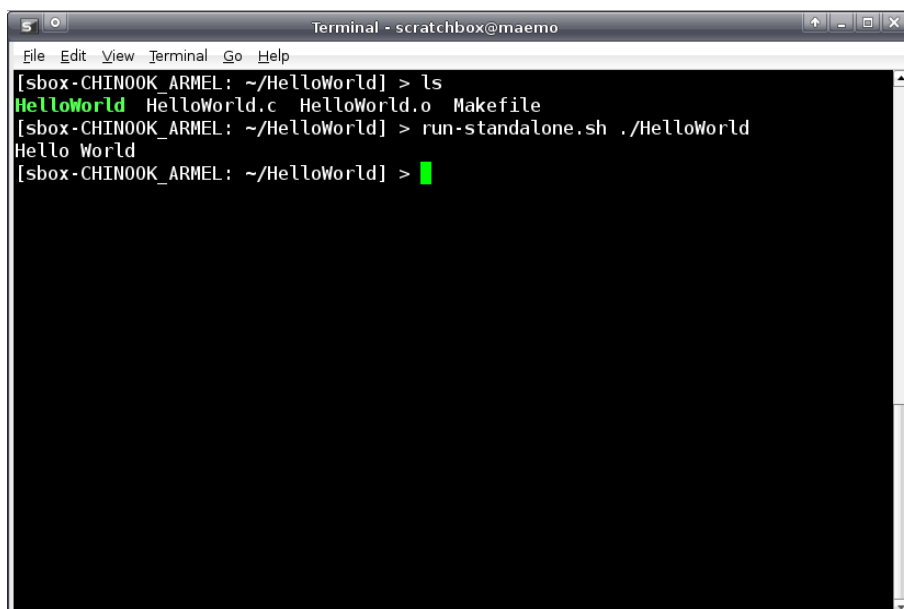
Figura A.4: Elección del target

A terminal window titled "Terminal - scratchbox@maemo" with a menu bar (File, Edit, View, Terminal, Go, Help). The terminal shows the following commands and output:

```
[sbox-CHINOOK_ARMEL: ~/HelloWorld] > ls
HelloWorld.c  Makefile
[sbox-CHINOOK_ARMEL: ~/HelloWorld] > make
gcc `pkg-config --cflags hildon-1 gtk+-2.0` HelloWorld.c -c
gcc `pkg-config --libs hildon-1 gtk+-2.0` HelloWorld.o -o HelloWorld

to run the compiled binary:
$ run-standalone.sh ./HelloWorld
[sbox-CHINOOK_ARMEL: ~/HelloWorld] > ls
HelloWorld HelloWorld.c HelloWorld.o Makefile
[sbox-CHINOOK_ARMEL: ~/HelloWorld] > █
```

Figura A.5: Compilación con *make*

A terminal window titled "Terminal - scratchbox@maemo" with a menu bar (File, Edit, View, Terminal, Go, Help). The terminal shows the following commands and output:

```
[sbox-CHINOOK_ARMEL: ~/HelloWorld] > ls
HelloWorld HelloWorld.c HelloWorld.o Makefile
[sbox-CHINOOK_ARMEL: ~/HelloWorld] > run-standalone.sh ./HelloWorld
Hello World
[sbox-CHINOOK_ARMEL: ~/HelloWorld] > █
```

Figura A.6: Ejecución con *run-standalone.sh ./HelloWorld*

Apéndice B

Guía de Nokia Energy Profiler v1.0 para Symbian

Introducción

Nokia Energy Profiler es una aplicación de medición para los móviles Nokia S60 3ª Edición, los FP1 hacia adelante. La aplicación permite a los desarrolladores probar y supervisar el consumo del dispositivo en tiempo real y consiste en un fichero SIS para ser instalado en el dispositivo móvil. La herramienta se puede instalar en todos los S60 3ª Edición, aunque el modo de grabación solo funciona en FP1 (Feature Pack 1) hacia adelante.

Instalación

El paquete de instalación (SIS) se pueden transferir a un dispositivo en múltiples formas diferentes, por ejemplo, usando USB, Bluetooth, MMC, o descarga directa desde Internet. En este ejemplo, la aplicación se ha instalado utilizando un cable USB y Nokia PC Suite. Si no tiene instalado PC Suite, se puede utilizar métodos alternativos o se puede descargar e instalar PC Suite desde las páginas de soporte de producto de Nokia:

1. Conectar el dispositivo a un PC mediante un cable USB.
2. Seleccionar el modo **PC Suite** para la conexión USB.
3. Hacer doble clic en el paquete SIS del PC Suite.
4. Seleccionar **Yes** para la confirmación de la instalación en el PC.

Nokia PC Suite transfiere el paquete SIS automáticamente al dispositivo y se inicia el procedimiento de instalación.

5. Seleccionar **Yes** a la confirmación de la instalación en la pantalla del dispositivo.
6. Seleccionar **Continue** en la pantalla de Detalles.
7. Esperar a que se instale la aplicación. Se podrá ver una barra de progreso en la pantalla.

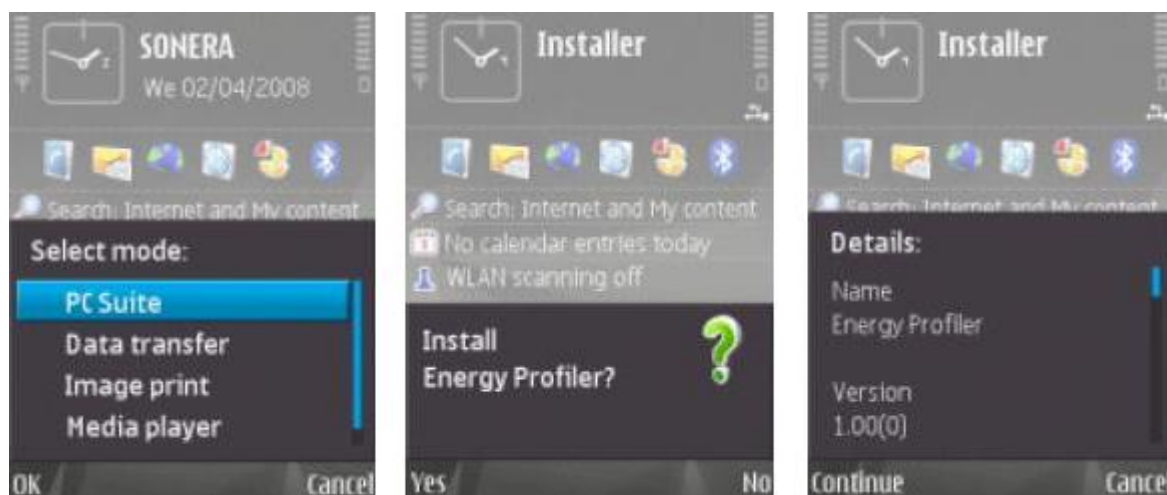


Figura B.1: Instalación en Symbian Nokia Energy Profiler

Después de la instalación, el dispositivo regresa a la situación en la que se encontraba antes de comenzar. También se puede ver una nota de confirmación en el Pc “Installation complete”, y se puede desconectar el cable USB.

Ahora se está listo para ejecutar la aplicación Nokia Energy Profiler.

Uso

La aplicación Nokia Energy Profiler se encuentra instalado en el carpeta de las aplicaciones del dispositivo.

1. Arrancar la aplicación seleccionando el icono.
2. Seleccionar **Options** >> **Start** para comenzar a registrar el consumo del dispositivo. Tambien se puede empezar a tomar medidas usando la tecla “2”.

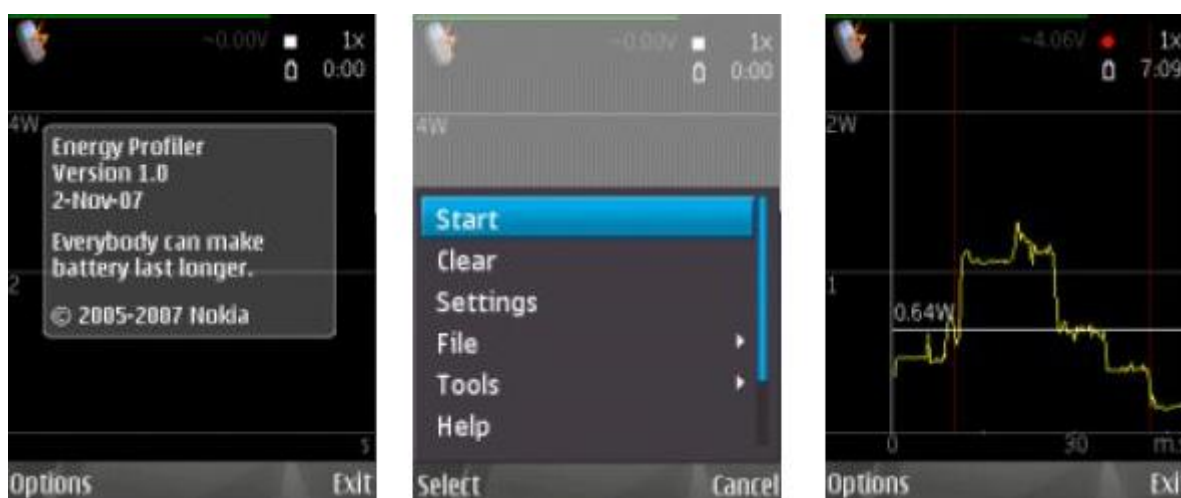


Figura B.2: Uso en Symbian Nokia Energy Profiler

En la pantalla se podrá ver un punto rojo que parpadea y una gráfica que indica el consumo. Pulsar tecla de “Arriba/Abajo” para cambiar la escala vertical. Pulsando la tecla “End” se envía la aplicación a segundo plano, pudiendo volver a ella dejando pulsado la tecla de “Aplicaciones”.

Consejos

- Las capturas de pantalla son automáticas. Se puede deshabilitar y usar 1(o 5 sg) como periodo de medición para obtener resultados mas precisos en el modo de espera.
- No dejar corriendo aplicaciones innecesarias durante la toma de medidas. Las aplicaciones que se quedan en segundo plano pueden afectar a los resultados que se obtienen puesto que Nokia Energy Profiler toma medidas de la potencia total de uso en el dispositivo.

- Mediante las capturas de pantalla se pueden guardar y obtener partes de la traza. Ver **Options >> Seetings** y seleccionar **Screenshots** del menú. Insertar marcadores en la traza dejando pulsada la tecla “5”.
- No dejar el cargador conectado al dispositivo mientras las mediciones, ya que produce resultados erróneos. El gráfico de fondo es gris mientras que el cargador esté conectado.
- Colocar Nokia Energy Profiler como primera aplicación en espera activa. De esta forma se puede cambiar rápidamente a la aplicación.
- Tenga en cuenta que obtiene más zoom vertical si deja pulsado el botón de arriba/abajo en los extremos normales del zoom.
- Aunque la capacidad de la batería se lee normalmente, es conveniente comprobar porque es posible que de valores erróneos de mAh.

Después de terminar las mediciones, volver a la aplicación Nokia Energy Profiler y seleccionar **Options >> Stop** para parar. Salvar los resultados seleccionando **Options >> File >> Save**.

Ahora, puede ver los resultados en la pantalla de la aplicación, o exportarlos en distintos formatos para su posterior análisis, seleccionando **Options >> File >> Export**.

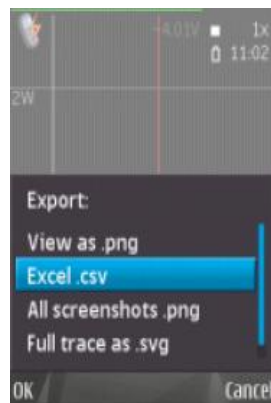


Figura B.3: Fromatos para exportar

	A	B	C	D	E
1	TIMESTAMP	POWER	CURRENT	VOLTAGE	TEMP
2	0.106	0.394	0.12	4.058	30.8
3	0.356	0.394	0.097	4.058	30.8
4	0.606	0.402	0.099	4.058	30.8
5	0.857	0.381	0.094	4.058	30.8
6	1.355	0.39	0.096	4.058	30.8
7	1.606	0.386	0.095	4.058	30.8
8	1.856	0.39	0.096	4.058	30.8
9	2.106	0.39	0.096	4.058	30.8
10	2.606	0.39	0.096	4.058	30.8
11	2.856	0.39	0.096	4.058	30.8
12	3.355	0.377	0.093	4.058	30.8
13	3.605	0.39	0.096	4.058	30.8
14	3.855	0.39	0.096	4.058	30.8
15	4.355	0.414	0.102	4.058	30.8

Figura B.4: Archivo exportado (.csv) visualizado con Excel

Teclas de acceso rápido

2	Comenzar/Parar las medidas
5(dejando pulsado)	Insertar marcadores
7(dejando pulsado)	Rotar el display
8	
0	Alternar entre el tipo de medida (Potencia, Corriente, Voltaje, Temperatura)
0(dejando pulsado)	Entrar y salir del modo de visualización
*/#	Acercar/Alejar respecto al eje del tiempo
Arriba/Abajo	Acercar/Alejar respecto al eje de la potencia
Izquierda/Derecha	Desplazarse
Tecla del centro(dejando pulsado)	Ver pantallazos (screenshot)
Tecla de terminar llamada	Envía la aplicación a 2º plano

Iconos de la pantalla

A continuación se muestran tres capturas de pantalla de la aplicación Nokia Energy Profiler en el móvil N81.

~ 3.97V	Media del voltaje durante la medición (Figura 9)
3.94V	Instantanea del voltaje (Figuras 10 y 11)
1x	Nivel del zoom.
8:17	Tiempo estimado en hr:min que le queda a la batería llena con el promedio medido el consumo de energía.
Cuadro Blanco	Indica el modo de visualización.
Línea gris vertical	Indica el inicio o el final de la región de las medidas tomadas.
Línea roja vertical	Indica la región del screenshot que se puede ver en el modo de visualización.
Línea verde(parte superior)	Memoria disponible en la memoria del teléfono o en una tarjeta. El ancho total corresponde a 64Mbytes.

Por favor, leer **Options >> Help** porque hay más información útil.

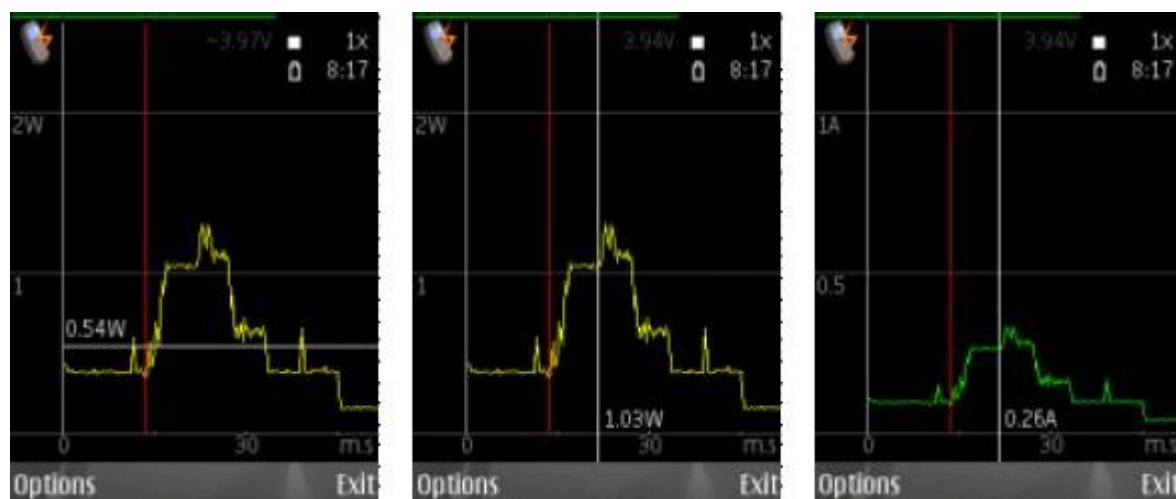


Figura B.5: Iconos de la pantalla de Symbian Nokia Energy Profiler

Apéndice C

Programa Ttcp

TTCP, es un programa basado en sockets que mide el rendimiento de procesamiento de la red entre dos sistemas. Originalmente fue desarrollado para el sistema operativo BSD en 1984.

Este apéndice muestra un ejemplo sencillo utilizando el protocolo TCP y direcciones unicast:

1. Recepción: Para ello es necesario lanzar el proceso en modo de recepción. En nuestro caso, cuando se desee transmitir desde el N810, este proceso se lanzará en el ordenador que forma parte del escenario de medidas descrito en el capítulo 3. Esto se hace lanzando el programa `ttcp` con las siguientes opciones:

`ttcp -r -s -pport`

Donde:

- `-r`, indica que el programa se lanza en modo de recepción.
- `-s`, los datos recibidos se deshechan.
- `-p`, indica el puerto en el que se va a poner a la escucha el programa.

Ejemplo:

```
bash-2.01\$ ttcp -r -s -p5000
ttcp-r: buflen=8192, nbuf=2048, align=16384/+0, port=5000  tcp
ttcp-r: socket
```

Por defecto el programa se lanza con los siguientes parámetros:

- Protocolo TCP. En el caso de utilizar UDP, se necesita la opción “-u”.
- Tamaño de buffer 8192 bytes de datos.
- Número de buffers 2048. Este valor se puede modificar con la opción “-n numbufs”.
- Alineación del comienzo de los buffers 16384.
- Desplazamiento respecto del comienzo del buffer 0.

2. Transmisión: Desde el N810 se lanza el proceso de transmitir:

```
ttcp -t -s -pport host
```

Donde:

- -t, indica que ttcp se lanza en modo de transmisión
- -s, impide que los datos a enviar se tomen de la entrada estándar
- -pport, especifica el puerto de la máquina remota a la que se envían los datos
- host, indica el nombre del equipo de destino.

Ejemplo:

```
bash-2.01\$ ttcp -t -s -p5000 192.168.1.82
ttcp-t: buflen=8192, nbuf=2048, align=16384/+0, port=5000 tcp ->
192.168.1.82
ttcp-t: socket
ttcp-t: connect
ttcp-t: 16777216 bytes in 0.36 real seconds = 45145.06 KB/sec +++
ttcp-t: 2048 I/O calls, msec/call = 0.18, calls/sec = 5643.13
ttcp-t: 0.0user 0.1sys 0:00real 50% 0i+0d 0maxrss 0+2pf 0+0csw
```

Esto significa que se ha abierto un socket en el sistema, que se ha establecido conexión, el número de bytes de datos transmitidos, el tiempo de transmisión y la velocidad de transmisión resultante(Kb).

Mientras que en recepción ocurre lo siguiente:

```
bash-2.01\$ ttcp -r -s -p5000
ttcp-r: buflen=8192, nbuf=2048, align=16384/+0, port=5000 tcp
ttcp-r: socket
ttcp-r: accept from 192.168.1.100
ttcp-r: 16777216 bytes in 0.38 real seconds = 42749.72 KB/sec +++
ttcp-r: 2049 I/O calls, msec/call = 0.19, calls/sec = 5346.32
ttcp-r: 0.0user 0.1sys 0:00real 34% 0i+0d 0maxrss 0+2pf 0+0csw
```


Pruebas con ttcp en Maemo

A continuación se muestra un ejemplo de comunicación de los experimentos del capítulo 4 entre el N810 y el ordenador, son con UDP, y “-w” (tiempo entre paquetes):

Ordenador:

```
@dahl:~$ ./ttcp -r -u -s -l30 -n17067 -p5000
ttcp-r: buflen=30, nbuf=17067, align=16384/0, port=5000  udp
ttcp-r: socket
ttcp-r: 511980 bytes in 713.44 real seconds = 0.70 KB/sec +++
ttcp-r: 17068 I/O calls, msec/call = 42.80, calls/sec = 23.92
ttcp-r: 0.0user 0.1sys 11:53real 0% 0i+0d 0maxrss 0+0pf 17067+20csw
```

Maemo:

```
~$ ./ttcp -t -u -s -l30 -n17067 -w35 -p5000 192.168.1.82
ttcp-t: buflen=30, nbuf=17067, align=16384/0, port=5000  udp -> 192.168.1.82
ttcp-t: socket
ttcp-t: 512010 bytes in 149.53 real seconds = 3.34 KB/ sec +++
ttcp-t: 17073 I/O calls, msec/call = 8.97, calls/sec = 114.18
ttcp-t: 0.0user 0.1sys 2:29real 0% 0i+0d 0maxrss 0+2pf 17068+17134csw
```

Captura por tcpdump (*tcpdump -i eth1 “udp”*) lanzada en el ordenador:

```
18:46:09.237242 IP 192.168.1.101.49152 > 192.168.1.82.5000: UDP, length 30
18:46:09.276925 IP 192.168.1.101.49152 > 192.168.1.82.5000: UDP, length 30
18:46:09.315375 IP 192.168.1.101.49152 > 192.168.1.82.5000: UDP, length 30
18:46:09.354392 IP 192.168.1.101.49152 > 192.168.1.82.5000: UDP, length 30
18:46:09.393392 IP 192.168.1.101.49152 > 192.168.1.82.5000: UDP, length 30
18:46:09.432560 IP 192.168.1.101.49152 > 192.168.1.82.5000: UDP, length 30
18:46:09.471503 IP 192.168.1.101.49152 > 192.168.1.82.5000: UDP, length 30
```


Apéndice D

Flasheo e instalación de paquetes en Nokia N810 Internet Tablet

Esta sección presenta una descripción de cómo flashear el dispositivo N810, desde Linux y Windows, además de las distintas formas de instalar paquetes en él.

FLASHEO

Para ambos sistemas operativos, existen una serie de pasos que se pueden consultar en los siguientes enlaces:

Linux http://maemo.org/community/wiki/HOWTO_FlashLatestNokiaImageWithLinux?highlight=%2528CategoryFlashing%2529

Windows <http://www.nokia.es/A4886330>

Linux

- En primer lugar es necesario la imagen y el binario.
- Es necesario ser root del PC desde donde se va a flashear.
- El siguiente paso es conectar el dispositivo al PC mediante un cable USB, y abrir una consola en el PC:

```
sudo ./flasher -F RX34+RX44\_DIABLO\_0.2008.12-14\_RD\_LEGACY\_MRO\_ARM.bin -f -R
```

donde:
flasher, es el binario
RX34..., es la imagen

- De esta manera, se tienen que ver los siguientes mensajes,

Ordenador:
flasher v2.2.4 (Mar 13 2008)
Suitable USB device not found, waiting

N810:
USB device found found at bus 001, device address 007
Found device RX-44, hardware revision 0805
NOLO version 1.1.7
Version of 'sw-release': RX-44_2008SE_2.2007.50-2_PR_MR0
Sending xloader image (9 kB)...
100 % (9 of 9 kB, avg. 900 kB/s)
Sending secondary image (97 kB)...
100 % (97 of 97 kB, avg. 977 kB/s)
Flashing bootloader... done.
Sending kernel image (1506 kB)...
100 % (1506 of 1506 kB, avg. 959 kB/s)
Flashing kernel... done.
Sending initfs image (3153 kB)...
100 % (3153 of 3153 kB, avg. 957 kB/s)
Flashing initfs... done.
Sending and flashing rootfs image (196352 kB)...
100 % (196352 of 196352 kB, avg. 935 kB/s)
Finishing flashing... done

- A continuación, el dispositivo se reinicia solo. Al terminar, se desconecta del ordenador.

Windows

- Al igual que con Linux, es necesario la imagen, pero además, es necesario un asistente para actualizar el software que se descarga desde:
<http://www.nokia.es/A4843574>
- Se conecta el dispositivo al ordenador mediante el cable USB.

- Se arranca el asistente de actualizar el software y se siguen las instrucciones que aparecen en pantalla, seleccionando la imagen que tenemos previamente guardada en el ordenador.
- Por último, tras reiniciarse el N810, se desconecta del ordenador.

En ambos casos, cuando el dispositivo se ha flasheado correctamente y una vez arrancado, para terminar la configuración, aparecen las siguientes pantallas que no son obligatorias porque se pueden visualizar más tarde:

- start-up: Se selecciona el lenguaje, la hora y el día/mes/año.
- phone-selection: En esta pantalla se selecciona un teléfono para utilizar a través de bluetooth.
- operator setup: Se selecciona un operador para el N810.

INSTALACIÓN DE PAQUETES

Hay tres maneras de instalar paquetes en el dispositivo N810:

- Mediante el PC
 - Descargar los paquetes y almacenarlos en el ordenador.
 - Conectar el dispositivo a través de USB.
 - Guardar los paquetes en el dispositivo, por ejemplo en la memoria interna.
 - Buscar los paquetes desde el dispositivo e instalarlos mediante “Application Manager”.
- Mediante el navegador web desde el N810
 - Buscar los paquetes a través de la web y se guardan en el dispositivo. Por ejemplo desde esta página: <http://maemo.org/community/wiki/ApplicationCatalog2006/>
 - Buscar los paquetes guardados e instalarlos mediante “Application Manager”.
- Mediante “Application Manager”
 - Seleccionar *Browse installable applications*.

- Al seleccionar una categoría, se despliega una lista de paquetes mostrando el nombre, la versión y el tamaño. Al seleccionar uno de ellos, se muestra una pequeña descripción.
- Seleccionar e instalar.

Además, incorpora la opción de buscar un determinado paquete, mostrar nuevas versiones de los que se han instalado previamente.

Para más información, ir a *Options > Help* en el menú de la aplicación

Apéndice E

Cómo realizar las medidas en el Nokia N810 Internet Tablet

En esta sección se muestra un ejemplo de los pasos que se dan para realizar las medidas, cómo se generan los datos, y cómo se extraen para luego generar las gráficas.

Los pasos que se dan son los siguientes:

1. Antes de empezar, Bluetooth y display en Off.
2. Se conecta el N810 a la red inalámbrica del escenario.
3. Se abre la consola en el ordenador y se escribe la siguiente secuencia con el programa `ttcp` para recibir datos:

```
./ttcp -r -u -s -n17067 -l600 -p5000
```

4. Se abre la consola en el N810 y se escribe lo siguiente:

```
./ttcp -t -u -s -n17067 -l600 -w35000 -p5000 192.168.1.82
```

donde los últimos números corresponden a la dirección IP de destino.

5. Se abre el programa Nokia Energy Profiler y se empiezan a tomar medidas.
6. En la consola del N810 ejecutar la secuencia escrita (paso 4).
7. Una vez termine el proceso, parar el programa que está tomando las medidas, y guardar y exportar los datos.

8. Pasar los datos al PC, y abrir el archivo **.csv** que contiene los datos tomados.

- En el archivo, separar los datos con coma y guardar. El segundo valor corresponde a la potencia consumida en cada instante.
- Se abre con el programa excel y se calcula el eje del tiempo:
El primer dato corresponde al instante 0s. A los demás datos hay que restarle el primer valor y dividirlo entre 1000 porque se representa en “ms” (ver figura E.1). Luego, se pasa la columna C al lugar de la A, de tal forma que A=Tiempo, y B=Potencia.

	A	B	C	D
1	1853282024	0.87	$=(a1-1853282024)/1000$	
2	1853283024	0.81		
3	1853284026	0.40		
4	1853285026	0.31		
5	1853286026	0.33		
6	1853287026	0.32		
7	1853288026	0.31		
8	1853289026	0.31		
9	1853290026	0.31		
10	1853291024	0.71		
11	1853292023	1.41		
12	1853293024	1.65		
13	1853294026	1.41		
14	1853295025	1.15		
15	1853296026	1.16		
16	1853297026	1.15		
17	1853298026	1.16		
18	1853299026	1.16		

Figura E.1: Maemo, calcular eje del tiempo

9. Guardar documento y cambiarle el formato a **.dat**.

10. Preparalo para utilizarlo con los programas R¹ y gnuplot² para el análisis.

¹Es un lenguaje y un entorno para computación y gráficos estadísticos. Ofrece una gran variedad de estadísticas (modelo lineal y no lineal, tests clásicos de estadísticas, análisis de series temporales, clasificación, agrupación, ...) y técnicas gráficas.

²Es un programa en línea de comandos que permite dibujar gráficas de funciones en 2 y 3 dimensiones a través de las fórmulas que las definen. También puede dibujar gráficos usando una tabla de coordenadas (en formato sólo texto) creadas con cualquier programa.

Apéndice F

Glosario de Términos

AAC: *Advanced Audio Coding*, formato informático de señal digital audio basado en el Algoritmo de compresión con pérdida, un proceso por el que se eliminan algunos de los datos de audio para poder obtener el mayor grado de compresión posible, resultando en un archivo de salida que suena lo más parecido posible al original.

ACK: *ACKNOWLEDGEMENT*, en comunicaciones entre dispositivos, mensaje que se envía para confirmar que un mensaje o un conjunto de mensajes han llegado.

AJAX: *Asynchronous JavaScript And XML*, técnica de desarrollo web para crear aplicaciones interactivas.

AMR: *Adaptive Multi-Rate*, formato de compresión de audio optimizado para la codificación de voz

AODV *Ad hoc On-Demand Distance Vector*, protocolo de enrutamiento para redes móviles ad hoc (manet) y otras redes inalámbricas ad hoc.

APSD: *Automatic Power Save Delivery*, especificación de la red 802.11e que facilita el ahorro de energía. Solo afecta al uso de la radio y a la batería si el cliente wireless también lo soporta.

ARM: *Advanced RISC Machines*, familia de microprocesadores RISC diseñados por la empresa Acorn Computers y desarrollados por Advanced RISC Machines Ltd.

Atom: Fichero en formato XML usado para Redifusión web.

ASX: *Advanced Stream Redirector*, también conocidos como metarchivos de Windows Media, son archivos de texto que proporcionan información acerca de una secuencia de archivos y de su presentación.

AVI: *Audio Video Interleave*, formato de archivo contenedor de audio y vídeo.

AWB: Fichero de audio del RealPlayer.

BMP: *BitMaP*, extensión de archivos de imagen del formato llamado Mapa de Bits. Poseen una compresión sin pérdida de calidad y suelen ocupar mucho espacio de almacenamiento.

BSD: *Berkeley Software Distribution*, sistema operativo UNIX desarrollado y distribuido por la Computer Systems Research Group (CSRG) de la Universidad de California, Berkeley, 1977 a 1995.

BSSID: *Basic Service Set Identifier*, identificación de redes inalámbricas en una red ad-hoc.

CMOS: *Complementary Metal Oxide Semiconductor*, una de las familias lógicas empleadas en la fabricación de circuitos integrado.

COITT: *Colegio Oficial de Ingenieros Técnicos de Telecomunicación*.

DCCP *Datagram Congestion Control Protocol*, protocolo de nivel de Transporte orientado a mensaje.

DDR RAM: *Double Data Rate Random-Access Memory*, módulos de memoria RAM, que permite la transferencia de datos por dos canales distintos simultáneamente en un mismo ciclo de reloj. A su vez, es la memoria desde donde el procesador recibe las instrucciones y guarda los resultados.

DBUS: *Desktop Bus*, sistema de comunicación entre procesos (IPC), para aplicaciones de software con el fin de comunicarse entre sí.

DHCP: *Dynamic Host Configuration Protocol*, protocolo de red de tipo cliente/servidor que permite a los nodos de una red IP obtener sus parámetros de configuración automáticamente.

DSR: *Dynamic Source Routing*, protocolo de enrutamiento para redes inalámbricas.

DTIM: *Delivery Traffic Indication Message*, establece el intervalo de reactivación para clientes que se encuentren en modo de ahorro de energía.

- EDGE:** *Enhanced Data rates for GSM of Evolution*, también conocida como EGPRS, tecnología de la telefonía móvil, que actúa como puente entre las redes 2G y 3G. Su objetivo es aumentar las velocidades de transmisión de datos y la eficiencia del espectro y facilitar las nuevas aplicaciones y el aumento de capacidad para el uso móvil.
- EDR:** *Enhanced Data Rate*, técnica incorporada en la versión 2.0 de Bluetooth que permite mejorar las velocidades de transmisión en hasta 3Mbps.
- Firefox:** *Mozilla Firefox*, navegador multiplataforma descendiente de Mozilla Application Suite. Está disponible en varias versiones de Microsoft Windows, Mac OS X, GNU/Linux y algunos sistemas basados en Unix. Su código fuente es software libre.
- FTP:** *File Transfer Protocol*, protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP basado en la arquitectura cliente-servidor.
- GIF:** *Graphics Interchange Format*, extensión y formato de imagen creado por CompuServe en 1987 soportable en múltiples plataformas. Suele utilizarse para gráficos, especialmente pequeños, que pueden ser animados, con un máximo de 256 colores.
- GNOME:** *GNU Network Object Model Environment*, entorno de escritorio e infraestructura de desarrollo para sistemas operativos Unix y derivados Unix compuesto enteramente de software libre; forma parte del proyecto GNU.
- GNU:** *acrónimo recursivo de ¡GNU No es Unix!*. El proyecto GNU se inició en 1984 con el objetivo de crear un sistema operativo completo tipo Unix de software libre, sin embargo el núcleo de GNU no está finalizado, así que se usa GNU con el núcleo Linux. La combinación de GNU y Linux es el sistema operativo GNU/Linux.
- GPS:** *Global Positioning System*, sistema global de navegación por satélite que permite determinar en todo el mundo la posición de un objeto, persona,... Funciona mediante una red de satélites en órbita sobre el globo terrestre, el receptor localiza como mínimo tres de ellos y recibe unas señales indicando la posición y reloj de cada uno de ellos, para calcular con triangulación la posición donde se encuentra.
- GSM:** *Groupe Special Mobile*, estándar para la comunicación mediante teléfonos móviles que incorporan tecnología digital. GSM se considera, por su velocidad de transmisión y otras características, un estándar de segunda generación (2G).

GTK: Biblioteca del equipo GTK+, la cual contiene los objetos y funciones para crear la interfaz gráfica de usuario. A su vez, GTK+, *The GIMP Toolkit*, es un conjunto de bibliotecas multiplataforma para desarrollar interfaces gráficas de usuario (GUI), principalmente para los entornos gráficos GNOME, XFCE y ROX aunque también se puede usar en el escritorio de Windows, MacOS y otros.

HID: *Human Interface Devices*, dispositivos que se conectan a ordenadores e interactúan con personas, como el teclado, ratón,...

HSDPA: *High Speed Downlink Packet Access*, optimización de la tecnología espectral UMTS/WCDMA. Consiste en un nuevo canal compartido en el enlace descendente que mejora significativamente la capacidad máxima de transferencia de información pudiéndose alcanzar tasas de hasta 14 Mbps. Soporta tasas de throughput promedio cercanas a 1 Mbps. Es la evolución de la tercera generación (3G) de tecnología móvil, llamada 3.5G, y se considera el paso previo antes de la cuarta generación (4G).

HW: *Hardware*, Conjunto de los componentes que integran la parte material de un dispositivo informático.

H263/H264: Estándar de códec de vídeo diseñado originalmente como un formato de baja tasa de bits comprimido. El siguiente paso al H263, es el H264, también conocido como AVC y MPEG-4 parte 10, proporcionando una mejora en la capacidad.

IEEE: *Institute of Electrical and Electronics Engineers*, asociación técnico-profesional mundial dedicada a la estandarización.

IFS: *Interframe Space*, tiempo de espera para enviar datos, utilizado en la interfaz 802.11.

IMAP: *Internet Message Access Protocol*, protocolo que permite a un cliente local acceder y manipular sus mensajes de correo electrónico en un servidor remoto. Proporciona una funcionalidad más rica y compleja que POP.

IP: *Internet Protocol*, protocolo no orientado a conexión usado para la comunicación de datos a través de una red de paquetes conmutados.

JavaScript: Lenguaje de scripting basado en Objetos, utilizado para acceder a objetos en aplicaciones, integrado en un navegador web permitiendo el desarrollo de interfaces de usuario mejoradas y páginas web dinámicas.

JPE: Formato gráfico de la familia de los JPEG. También es una extensión de archivo de imagen (".jpe").

JPEG: *Joint Photographic Experts Group*, algoritmo diseñado para comprimir imágenes fijas con 24 bits de profundidad o en escala de grises. La compresión es con pérdida de calidad, por lo tanto al descomprimir la imagen no es la misma. Los ficheros que utilizan esta compresión llevan la extensión ".jpe" o ".jpg".

LAN: *Local Area Network*, red de área local.

MAC: *Media Access Control*, número único de 48 bits asignado a cada tarjeta de red. Se conoce también como la dirección física para identificar dispositivos de red.

Maemo OS: *Maemo Operating system*, sistema operativo propietario diseñado para dispositivos móviles e internet tablets.

MID: *Mobile Internet Device*, dispositivo móvil para conexión a internet.

miniSD: *mini Secure Digital*, formato de tarjeta de memoria flash, con unas proporciones de 21.5mm x 20mm x 1.4mm.

microSD: *micro Secure Digital*, formato de tarjeta de memoria flash, más pequeña que la miniSD, con unas proporciones de 15mm x 11mm x 1mm.

MMS: *Multimedia Messaging System*, estándar de mensajería que le permite a los teléfonos móviles enviar y recibir contenidos multimedia, incorporando sonido, video, fotos. El límite de cada mensaje multimedia suele ser de 100 o 300 KB, dependiendo de cada móvil.

MPEG-1/MPEG-4: Grupo de estándares de codificación de audio y vídeo normalizados por el grupo MPEG (Moving Pictures Experts Group). MPEG-1 vídeo se utiliza en el formato Video CD. MPEG-4 toma muchas de las características de MPEG-1 y MPEG-2.

MP2: *MPEG-1 Audio Layer 2*, es una compresión de audio con pérdida. Aunque ha sido suplantado por mp3, se sigue utilizando en DAB y de televisión digital DVB-T.

MP3: Formato de audio que combina gran calidad de sonido y poco tamaño. Redujo el tamaño de los archivos de música conocidos hasta diez veces casi sin perder calidad por la compresión.

- MP4:** MP4 o MPEG-4 Part 14, estándar de formato multimedia que es parte del MPEG-4. Se usa especialmente para el almacenamiento de video y audio digital, especialmente los definidos por MPEG, pero también puede almacenar otros datos como subtítulos e imágenes.
- MTU:** *Maximum Transfer Unit*, indica el tamaño máximo en bytes que se pueden enviar en una comunicación de red sin que el paquete se fragmente.
- M3U:** *MPEG Versión 3.0 URL*, formato de archivo que almacena listas de reproducción de medios.
- M4A:** La extensión m4a ha sido popularizada por Apple, quien inició el uso de la extensión ".m4a" en su software "iTunes" para distinguir entre archivos MPEG-4 de audio y vídeo (M4A y M4V respectivamente).
- NIC:** *Network Interface Card*, chip de la tarjeta de red que se encarga de servir como interfaz de Ethernet entre el medio físico y el equipo.
- OMA DRM:** *Open Mobile Alliance Digital Rights Management*, organización de estándares que desarrolla estándares abiertos para la industria de telefonía móvil.
- OTA:** *Over The Air*,
- PDF:** *Portable Document Format*, formato de almacenamiento de documentos, desarrollado por la empresa Adobe Systems.
- PoC:** *Push to Talk over Cellular*, comunicación half duplex, que mientras que una persona habla, el otro(s) escucha. Similar a una característica de walkie-talkie que se proporciona a través de una red de telefonía móvil.
- POP3:** *Post Office Protocol*, protocolo de oficina de correos. Es un protocolo utilizado en clientes locales de correo para obtener los mensajes de correo electrónico almacenados en un servidor remoto.
- PNG:** *Portable Network Graphics*, formato gráfico comprimido sin pérdida de calidad. Sus archivos tienen la extensión ".png".
- PLS:** Formato de archivo multimedia de ordenador que almacena listas de reproducción.
- PSM:** *Power Save Mode*, modo ahorro de energía destinado a reducir el consumo energético de los dispositivos móviles.

- PS-STA:** *Power Save Station*, dispositivo de la interfaz 802.11 que utiliza ahorro de energía.
- RA:** *Real Audio*, formato de audio desarrollado por RealNetworks. Puede asociarse con el RealVideo para formar un archivo RealMedia. Utilizan la extensión “.ra” para sus archivos de audio.
- RSA:** *Iniciales de los apellidos de Ron Rivest, Adi Shamir y Leonard Adleman*, algoritmo asimétrico cifrador de bloques, que utiliza una clave pública, la cual se distribuye (en forma autenticada preferentemente), y otra privada, la cual es guardada en secreto por su propietario.
- RSS:** RSS 1.0 *RDF Site Summary* y RSS 2.0 *Really Simple Syndication*, familia de formatos de fuentes web codificados en XML. Se utiliza para suministrar a suscriptores de información actualizada frecuentemente
- RFC:** *Request For Comments*.
- RV:** *Real Video*, formato de video desarrollado por RealNetworks. RealMedia es adecuado como medio de streaming (aquel que puede ser visto mientras se envía a través de una red, como internet). Utiliza la extensión “.rm”.
- SDHC:** *Secure Digital High Capacity*, formato de tarjeta de memoria flash. Se trata de la versión 2.0 de las memorias SD. Son memorias de alta capacidad, que van de 4GB a 32GB.
- SDK:** *Software Development Kit*, conjunto de herramientas de desarrollo que le permite a un programador crear aplicaciones para un sistema concreto.
- SIM:** *Subscriber Identity Module*, tarjeta inteligente desmontable usada en teléfonos móviles que almacena de forma segura la clave de servicio del suscriptor usada para identificarse ante la red.
- SIP:** *Session Initiation Protocol*, protocolo de aplicación que pretende ser el estándar para la iniciación, modificación y finalización de sesiones interactivas de usuario, donde hay componentes como video, voz, juegos online, realidad virtual y mensajería instantánea.
- SMS:** *Short Message Service*, servicio disponible en los teléfonos móviles que permite el envío de mensajes cortos (también conocidos como mensajes de texto) entre teléfonos móviles, teléfonos fijos y otros dispositivos de mano. SMS fue diseñado originariamente como parte del estándar de telefonía móvil digital GSM, pero en la actualidad está disponible en una amplia variedad de redes, incluyendo las redes 3G.

- SMTP:** *Simple Mail Transfer Protocol*, protocolo para la transmisión de mensajes de correo electrónico en Internet.
- STA:** *Station*, dispositivo que tiene la capacidad para utilizar la interfaz 802.11.
- SSH:** *Secure Shell*, nombre de un protocolo y del programa que lo implementa, y sirve para acceder a máquinas remotas a través de una red.
- SVG:** *Scalable Vector Graphics*, especificación para describir gráficos vectoriales bidimensionales, tanto estáticos como animados, en formato XML.
- Symbian OS:** *Symbian Operating system*, sistema operativo propietario diseñado para dispositivos móviles.
- S60:** *60's serie*. Gama de dispositivos cuyo sistema operativo es Symbian.
- TCP:** *Transmission Control Protocol*, protocolo de comunicación orientado a conexión y fiable del nivel de transporte. Permite a dos máquinas establecer una conexión e intercambiar datos.
- TI OMAP:** *Texas Instruments Open Multimedia Application Platform*, familia de microprocesadores dirigido a portátiles y móviles con aplicaciones multimedia.
- TIF/TIFF:** *Tagged Image File Format*, formato de fichero para el almacenamiento de imágenes creado a mediados de los 80. No pierden calidad en compresión, permiten la máxima calidad en las imágenes.
- TFT:** *Thin Film Transistor*, tipo especial de transistor de efecto campo que se fabrica depositando finas películas de un semiconductor activo así como una capa de material dieléctrico y contactos metálicos sobre un sustrato de soporte.
- UAPSD:** *Unscheduled Automatic Power Save Delivery*, una característica de los dispositivos Wi-Fi que les permite ahorrar energía. Conocido también como WMM power save.
- UDP:** *User Datagram Protocol*, protocolo no orientado a conexión de la capa de transporte del modelo TCP/IP.
- UPnP:** *Universal Plug and Play*, arquitectura software abierta y distribuida que permite el intercambio de información y datos a los dispositivos conectados a una red.
- USB:** *Universal Serial Bus*, bus serie universal.

- VGA:** *Video Graphics Array*, pantalla de resolución 640x480.
- VMware:** Programa que simula un sistema físico (un ordenador, un hardware) con unas características de hardware determinadas. Un virtualizador por software permite ejecutar (simular) varios ordenadores (sistemas operativos) dentro de un mismo hardware de manera simultánea, permitiendo así el mayor aprovechamiento de recursos.
- WAV:** *WAVEform audio format*, formato de audio digital normalmente sin compresión de datos desarrollado y propiedad de Microsoft y de IBM que se utiliza para almacenar sonidos en el PC, admite archivos mono y estéreo a diversas resoluciones y velocidades de muestreo, su extensión es “.wav”.
- WAX:** Lista de reproducción de metarchivos que se puede utilizar con los archivos de Windows Media que tengan la extensión de archivo “.wma”.
- Wi-Fi:** Nombre de tecnología de comunicación
- WIMAX:** *Worldwide Interoperability for Microwave Access*, interoperabilidad mundial para acceso por microondas.
- WBMP:** *Wireless Application Protocol Bitmap Format*, también conocido como Wireless Bitmap, un formato de archivo de imágenes monocromo optimizadas para el uso en dispositivos móviles.
- WCDMA:** *Wideband Code Division Multiple Access*, acceso múltiple por división de código de banda ancha.
- WLAN:** *Wireless Local Area Network*, sistema de comunicación de datos inalámbrico.
- WMA:** *Windows Media Audio*, formato y extensión del mismo desarrollado por Microsoft para codificar archivos de audio digital similar al MP3, aunque pueden comprimir archivos con un ratio más alto que estos.
- WMDRM:** *Windows Media Digital Rights Management*, servicio de gestión de derechos digitales para la plataforma de Windows Media.
- WMM:** *Wi-Fi Multimedia*, tecnología que mantiene la prioridad en audio, video y voz, y deja otras aplicaciones y el tráfico con un funcionamiento mas lento.
- WMV:** *Windows Media Video*, conjunto de algoritmos para la compresión de videos, propiedad de Microsoft. Por lo general suele combinarse con el formato de sonido WMA (Windows Media Audio). WMV desde la versión 7 utiliza su propia tecnología no estandarizada de MPEG-4. Este formato

puede ser reproducido por la mayoría de los reproductores y los archivos que lo utilizan suelen tener la extensión “.wmv”.

WPL: Lista de reproducción de cliente escritas en un formato patentado. Microsoft introdujo este formato de archivo en Reproductor de Windows Media 9 Series. Con el formato “.wpl” se pueden crear listas de reproducción dinámicas, a diferencia de los formatos “.asx” y “.m3u”, con los que no se puede.

WVGA: *Wide Video Graphics Array*, resolución de pantalla de 800x480, 848x480 o 854x480.

WVX: Son metarchivos de Windows Media que hacen referencia a archivos de Vídeo de Windows Media “.wmv”.

xG: *x Generation*, generación x (2G, 3G).

3GP: *3rd Generation Partnership Project*, formato de archivos usado por teléfonos móviles para almacenar información de medios múltiples(audio y video). Es una versión simplificada del “ISO 14496-1 Media Format”, que es similar al formato de Quicktime.

Bibliografía

- [1] Maemo SDK. http://maemovmware.garage.maemo.org/2nd_edition/.
- [2] Standard for Information technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, 2007.
- [3] Comunidad Maemo. <http://maemo.org>, Abril 2008.
- [4] Introducción a Maemo. <http://maemo.org/intro/>, Marzo 2008.
- [5] Nokia N810 Internet Tablet. <http://www.nokia.es/productos/moviles/n810>, Marzo 2008.
- [6] Plataforma Maemo. <http://maemo.org/intro/platform/>, Septiembre 2008.
- [7] Especificaciones Nokia N810 Internet Tablet. <http://www.nokia.es/productos/moviles/n810/especificaciones>, Mayo 2009.
- [8] Especificaciones Nokia N95. <http://www.nokia-latinoamerica.com/productos/todos-los-modelos/nokia-n95/especificaciones-tecnicas>, Mayo 2009.
- [9] Forum Nokia. <http://www.forum.nokia.com>, Diciembre 2009.
- [10] Móviles Symbian. <http://www.symbian.org/devices>, Noviembre 2009.
- [11] Nokia 7650. <http://www.nokia.es/soporte/producto/7650>, Marzo 2009.
- [12] Nokia N770 Internet Tablet. <http://www.nokia.es/soporte/producto/770internettablet>, Marzo 2009.
- [13] Nokia N800 Internet Tablet. <http://www.nokia.es/soporte/producto/n800internettablet>, Marzo 2009.

- [14] Nokia N900. <http://www.nokia.es/productos/moviles/nokia-n900-maemo-pantalla-tactil-teclado-qwerty>, Marzo 2009.
- [15] Nokia N95. <http://www.nokia.es/soporte/producto/n95>, Marzo 2009.
- [16] Symbian Platform Releases. <http://www.symbian.org/symbian-feature-set/symbian-releases>, Diciembre 2009.
- [17] OVI. www.ovi.com/services/, Marzo 2010.
- [18] UIQ. <http://www.uiq.com>, Enero 2010.
- [19] Andrew S. Tanenbaum. *Redes de Computadoras*, chapter 6.4, pages 524–532. 2003.
- [20] Ashima Gupta and Prasant Mohapatra. Power consumption and conservation in wifi based phones: A measurement-based study. Marzo, 2007.
- [21] Energy Profiler, Symbian. Guía de Inicio Rápido. http://www.forum.nokia.com/main/resources/user_experience/power_management/nokia_energy_profiler, Octubre 2009.
- [22] L. Feeney. An energy-consumption model for performance analysis of routing protocols for mobile ad hoc networks. *Mobile Networks and Applications Journal*, 6(3):239–250, Junio, 2001.
- [23] L. Feeney and M. Nilson. Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. In *IEEE Conf on Computer Communications (Infocom'01)*, Abril, 2001.
- [24] Martin P. Clark. *Data Networks, IP and the Internet*, chapter 5, pages 165–209. 2003.
- [25] Matthew S.Gast. *802.11 Wireless Networks, The Definitive Guide*. 2002.
- [26] Information Sciences Institute University of Southern California. RFC 791, Internet Protocol. Septiembre, 1981.
- [27] J. Postel. RFC 768, User Datagram Protocol. 28 Agosoto, 1980.
- [28] Trevor Pering, Yuvraj Agarwal, Rajesh Gupta and Roy Want. Coolspots: Reducing the power consumption of wireless mobile devices with multiple radio interfaces. In *Proceedings of the 4th International Conference on Mobile Systems, Applications and Services (MobiSys'06)*, Junio, 2006.
- [29] William Stallings. *Comunicaciones y Redes de Computadores*, chapter 2.3, pages 29–39. 2004.